Development and Experimental Testing of a Speed Controlled PMSM Drive Using PSIM Visual Programming Environment

by

K.M.Narmada Damayanthi Ranaweera

Submitted to the Department of Electrical Engineering, Electronics, Computers and Systems
in partial fulfillment of the requirements for the degree of
Erasmus Mundus Joint Master Degree in Sustainable Transportation and Electrical Power Systems
at the
UNIVERSIDAD DE OVIEDO
August 2019

© Universidad de Oviedo 2019. All rights reserved.

Author . . . .
K.M.N.D.Ranaweera

Certified by . . . .
Giulio De Donato
Associate Professor
Thesis Supervisor

Development and Experimental Testing of a Speed Controlled PMSM Drive Using PSIM Visual Programming Environment

by

K.M.Narmada Damayanthi Ranaweera

Submitted to the Department of Electrical Engineering, Electronics, Computers and Systems
on August 31, 2019, in partial fulfillment of the
requirements for the degree of
Erasmus Mundus Joint Master Degree in Sustainable Transportation and Electrical Power Systems

Abstract

This paper presents the design and implementation of a speed controller for an Interior Permanent Magnet Synchronous Motor (IPMSM) using PSIM visual programming software environment based on vector control technique. A floating type TMS320F28335 Digital Signal Processor is used to realize the drive system. High Voltage Motor Control and Power Factor Correction kit, developed by Texas Instruments, is used in which the power and control blocks for the motor drive system are embedded. Theoretical analysis and design steps are elaborated along with explanations on the dynamic performance of the speed controller in the experimental results section.

Thesis Supervisor: Giulio De Donato
Title: Associate Professor

Acknowledgments

This thesis has been submitted in partial fulfillment of the degree of Erasmus Mundus Joint Masters Degree (EMJMD) in Sustainable Transportation and Electrical Power Systems (STEPS). Thesis work has been carried out at Sapienza University of Rome.

I would like to express my sincere gratitude to my supervisor, Associate Professor Giulio de Donato for giving me great inspiration, ideas, comments, and continuous support throughout the implementation of the project. It was a great opportunity to carry out my research work under his supervision.

I would also express my gratitude to all the Professors who taught me during my undergraduates and masters courses. I also owe thanks to my fellows, I had in the lab, during this thesis work for their valuable suggestions and help.

I would also like to thank my parents for their encouragement. Last but not least, I would like to thank my husband, sister, and colleagues who helped me in numerous ways to make this thesis a success.

Contents

List of Figures

List of Tables

List of Abbreviations

ADC Analog to Digital Converter. 43, 44, 67, 69, 74–77, 82

BLAC Brushless Alternative Current. 22, 23

BLDC Brushless Direct Current. 22, 23

CLTF Close Loop Transfer Function. 43

CSS Code Composer Studio. 74

DAC Digital to Analog Converter. 80, 81

DSP Digital Signal Processor. 75, 81

EMF Electro-Motive Force. 9, 21, 22, 65

FFT Fast Fourier Transform. 65

GPIO General Purpose Input Output. 71, 79

IPMSM Interior Permanent Magnet Synchronous Motor. 3, 19, 21, 58, 63, 64, 99

ISR Interrupt Service Routine. 77, 97

MTPA Maximum Torque per Ampere. 95, 112

OLTF Open Loop Transfer Function. 42, 43, 45, 61

PCB Printed Circuit Board. 67, 69

PFC Power Factor Correction. 20

PI Proportional and Integral. 40, 42, 47, 60, 91

PID Proportional, Integral and Derivative. 50

PMSM Permanent Magnet Synchronous Machines. 19–21, 23, 31

PWM Pulse Width Modulation. 25, 29, 71–74, 77, 92, 93

SPI Serial Peripheral Interface. 71, 79, 80

SPMSM Surface-mounted Permanent Magnet Synchronous Motor. 21, 64, 99

SPWM Sinusoidal Pulse Width Modulation. 25, 26

SVPWM Space Vector Pulse Width Modulation. 25, 26, 31

VSI Voltage Source Inverter. 25, 27, 41

VTO Vector Tracking Observer. 50, 52, 95, 96, 100, 109, 111

ZOH Zero Order Hold. 44, 61, 96

List of Symbols

δ_c Controller zero

δ_p Plant pole

λ_{pm} Permanent magnet flux linkage

ω Velocity of the synchronous reference frame

ω_e Electrical angular velocity

ω_m Mechanical angular velocity

τ Time constant

τ_p Time constant of speed loop

θ Angular position

θ_e Electrical Angle

f_{BW} Bandwidth of the controller

K_a Motor torque constant

K_D Derivative gain of the controller

K_I Integral gain of the controller

K_P Proportional gain of the controller

R_s Stator resistance per-phase

T	Sample time of speed loop
T_s	Sample time of current loop
T_{dt}	Dead time
B	Friction
J	Inertia of the machine
P	Number of polar pairs

Chapter 1

Introduction

Nowadays Permanent Magnet Synchronous Machines (PMSM) are used for a wide variety of residential and industrial drive applications, ranging from general-purpose drives to high-performance drives. They offer many advantages over the other types of motors. They have a higher efficiency than induction motors due to the absence of copper losses in the rotor. Moreover, due to the availability of cost-effective and strong magnet materials, PMSMs exhibit a high power density, high air-gap flux density and high torque-to-volume ratio.

PMSM is increasingly used in high-performance applications, such as robots and industrial machines, which require speed controllers with high accuracy [1]. Over the past few years, several control techniques have been developed for this purpose. Vector control technique is most commonly used due to its ability to achieve high-performance control characteristics and fast dynamic response.

The objective of the thesis is to design and analyze a speed control algorithm for an Interior Permanent Magnet Synchronous Motor (IPMSM) using PSIM software package based on vector control technique.

Chapter 1 presents a state of the art review on permanent magnet machine drives. Shaft position encoder technologies are described with a discussion on the limitations of these techniques. Conventional modulation techniques used for voltage source inverter are presented. Furthermore, vector control technique is briefly discussed.

Chapter 2 presents the theoretical modelling of PMSM in dq axis synchronous reference frame with a brief introduction to reference frame transformation.

Chapter 3 describes the theoretical analysis of current and speed control algorithms mainly focusing on the design and tuning of current and speed controllers. Designing of vector tracking observer to estimate speed from the encoder output is also described.

Chapter 4 describes details about the High Voltage Motor Control and Power Factor Correction (PFC) Kit which is the main hardware component in the experimental setup. Hardware and peripheral configuration are described for the hardware blocks used in the PSIM model. Furthermore, determining the parameters required for controller implementation is presented.

Chapter 5 presents the current and speed control loops modelling in PSIM software with a discussion of the initial rotor alignment procedure.

Chapter 6 presents the results of the experiments conducted to test the system. Finally, chapter 7 summarizes the work with a conclusion and provides suggestions for future work corresponds to the thesis.

1.1 Permanent Magnet Synchronous Motor

PMSM is constructed with stator phase windings and rotor permanent magnets. The air gap magnetic field provided by the permanent magnets is constant. Stator windings are commutated externally using a three-phase inverter topology. The structure of a PMSM is shown in Fig.1.1.1.

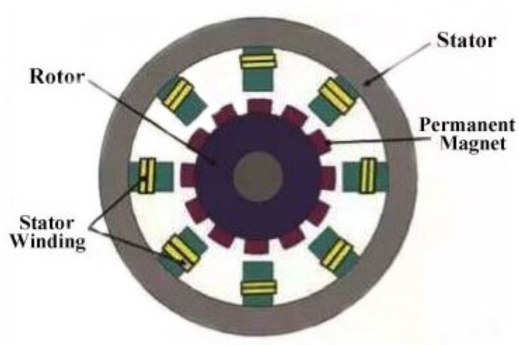

Figure 1.1.1: The Structure of a Permanent Magnet Synchronous Motor

Torque produced by the interaction of the stator and rotor magnetic fields becomes maximum when the magnetic vector of the rotor is perpendicular to the magnetic vector of the stator [2]. In addition to the electromagnetic torque, synchronous motors with a salient rotor exploit reluctance torque which can be utilized when designing speed controller.

PMSM can be classified into several categories which are described in the following subsection.

1.1.1 Classification of PMSMs

PMSMs can be classified into two categories based on the position of the permanent magnets as shown in Fig.1.1.1.1; Surface-mounted Permanent Magnet Synchronous Motor (SPMSM) in which magnets are mounted on the surface of the rotor and Interior Permanent Magnet Synchronous Motor (IPMSM) in which magnets are buried inside the rotor iron core.

IPMSM can be further categorized into two topologies according to the arrangement of the permanent magnets inside the rotor iron core. They are parallel topology and perpendicular topology shown in Fig.1.1.1.1.(B) and Fig.1.1.1.1.(C) respectively. IPMSMs with the parallel topology are more commonly used.

The perpendicular topology can concentrate the magnetic flux of the magnets and achieve a high pole number, thereby increasing flux density in the air gap. Thus, the motor can produce a high torque density even when low cost and low flux density material is used for magnets. However, flux concentration at the pole edges results in high harmonic distortion in the air-gap flux density near the pole tips. This causes harmonic distortion in the back Electro-Motive Force (EMF) profile which also leads to a high ripple in the torque produced. [3]

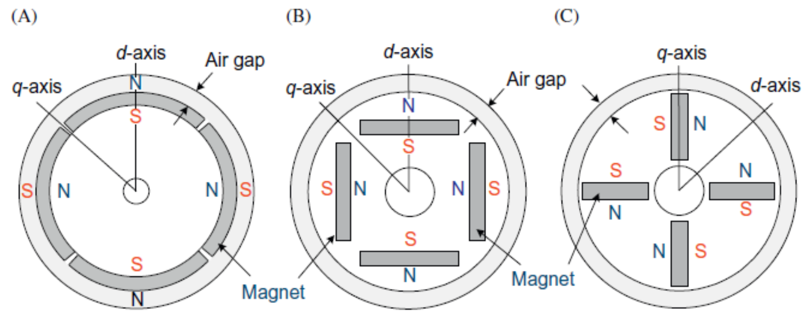

Figure 1.1.1.1: Rotor Topologies of PMSMs; (A) Surface mounted, (B) interior (parallel), and (C) interior (perpendicular) [3]

Permanent magnet machines can also be further classified into two categories according to the type of the back-emf waveform induced in the stator windings; Brushless Alternative Current (BLAC) and Brushless Direct Current (BLDC). In BLAC machines, sinusoidal back-emf is achieved by 120° magnet span and a sinusoidal current excitation. On the other hand in BLDC machines, trapezoidal back-emf is achieved by 180° magnet span and square wave current excitation. Excitation current and back-emf waveforms for the two types are shown in Fig.1.1.1.2.

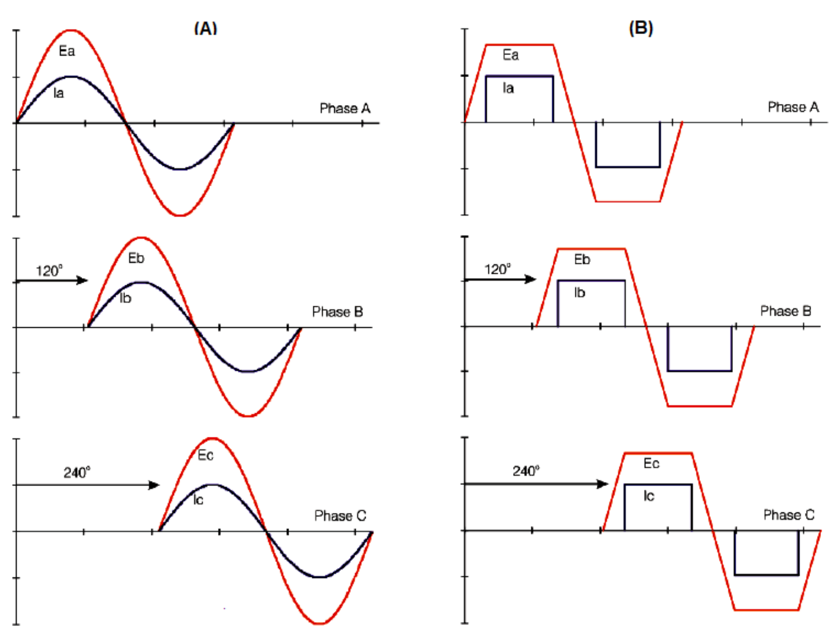

Figure 1.1.1.2: Current Excitation and Back-EMF Voltages; (A) BLAC, (B) BLDC [4]

BLDC machine has a higher torque density than BLAC due to better utilization of the magnetic circuit. However, BLDC output characteristics are inferior to BLAC drives in terms of torque and current smoothness. It has a high torque ripple resulting from back-emf and switching harmonics. [4]

1.2 Shaft Position Sensor

In order to develop control schemes for PMSMs, measurement of rotor position is required. A position sensor is mounted on the rotor shaft for this purpose. The most commonly used position measurement devices for motors are encoders and revolvers. A suitable position sensor with the required accuracy can be selected according to the application and performance desired by the motor.

The most commonly used type of encoder is the optical encoder, which consists of a rotating disk, a light source, and a photodetector. A disk with coded patterns of opaque and transparent sectors is mounted on the rotor shaft. As the disk rotates, these patterns interrupt the light emitted onto the photo detector, generating a digital pulse [5]. Fig.1.2.1. depicts two types of optical encoders; incremental encoder and absolute encoder which are discussed in the next subsections.

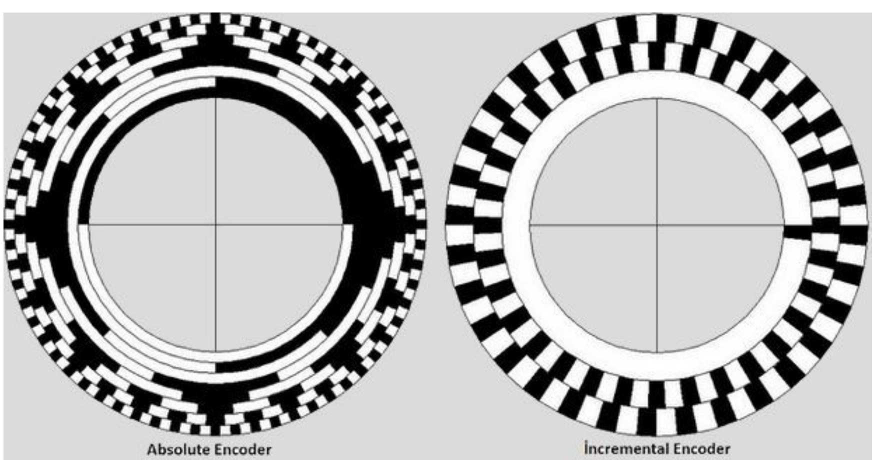

Figure 1.2.1: Disk Structure of Absolute and Incremental Encoder

1.2.1 Incremental Encoders

Incremental encoders are simple in design and cost-effective. The disk consists of two code tracks with sectors positioned 90^0 degrees out of phase. Incremental encoders are also called quadrature encoders for this reason. Two output channels, A and B are used to detect incremental angular motion. The direction of rotation is determined by checking whether the A channel signal leads or lags the B channel signal. Encoder resolution depends on the number of slots on a track. Falling and rising edges of the digital pulses A and B are detected and reading is updated four times during one pair of slot passage. Thus incremental encoder can have a maximum resolution equal to 4 times the number of slots of the encoder.

Most of the incremental encoders are also equipped with an extra output signal called the zero or index or reference signal, which supplies a single pulse per revolution. This single pulse is used for precise determination of a reference position. Fig.1.2.1.1. shows the generation of output pulses in an incremental encoder.

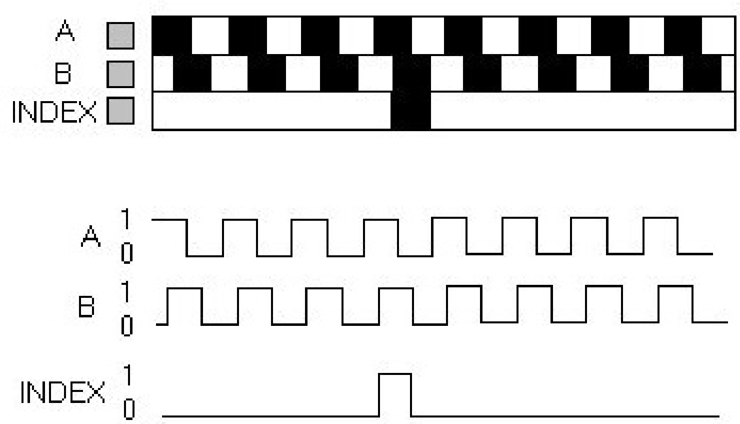

Figure 1.2.1.1: Pulse Patterns of an Incremental Encoder

1.2.2 Absolute Encoders

The absolute encoder detects the exact position of the rotor with a precision directly related to the number of bits of the encoder. It provides a digital output with a unique code pattern representing each position. This code is derived from independent tracks on the encoder disc (one for each "bit" of resolution) corresponding to individual photodetectors.

Absolute encoders are generally more expensive than incremental encoders. Because of their ability to provide absolute position of the rotor, they are used in applications where a device is inactive for long periods of time or moves at a slow rate. (ex: flood gate control, telescopes, cranes, etc..). They are also used in systems where position information must be retained after a power outage. [5]

1.3 Pulse Width Modulation Techniques

Pulse Width Modulation (PWM) is the most commonly used technique to control the output voltage of a Voltage Source Inverter (VSI). The objective of the Pulse Width Modulation (PWM) is to generate gating pulses for the inverter switches to produce an output voltage with the desired fundamental amplitude and frequency.

Two types of modulation techniques, Sinusoidal Pulse Width Modulation (SPWM) and Space Vector Pulse Width Modulation (SVPWM) are discussed in the next subsections.

1.3.1 Sinusoidal Pulse Width Modulation

SPWM is a modulation technique in which a sinusoidal signal is compared with the high-frequency triangular carrier wave to determine switching states of the inverter switches. If reference voltage V_{ref} is greater than carrier voltage V_c, the upper switch is turned on. When V_{ref} is lower than V_c, the lower switch is turned on. The peak-to-peak value of the triangular carrier wave is given as the DC-link voltage V_{dc}.

Fig.1.3.1.1. illustrates SPWM technique to generate inverter output voltage.

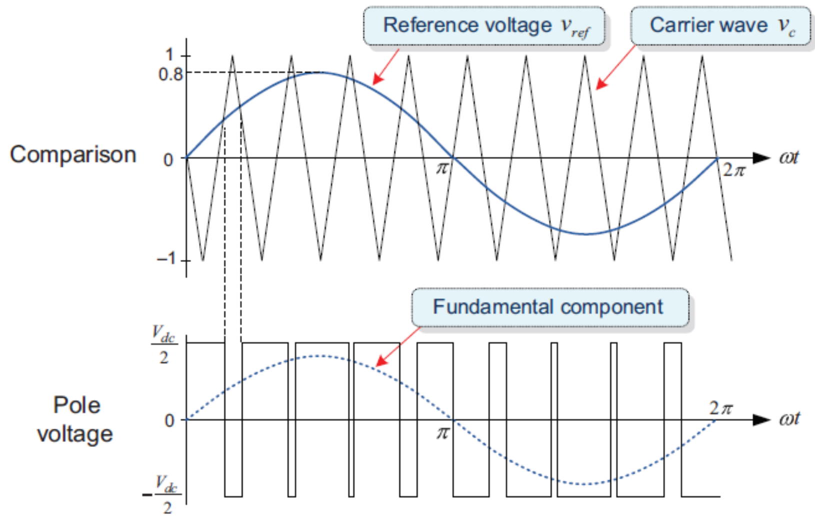

Figure 1.3.1.1: Voltage Waveform Generation using Sinusoidal PWM [3]

In this PWM technique, the necessary condition for linear modulation is that modulation index which is defined as, $V_{ref}/(V_{dc}/2)$ must be lower than 1.

The switching frequency of the inverter is equal to the frequency of the carrier wave. Therefore, this technique has the advantage of having a constant switching frequency. A higher switching frequency can be used in order to improve the quality of the voltage waveform. However, this leads to greater switching losses. Therefore, it is important to consider the overall performance of the system when selecting the switching frequency.

1.3.2 Space Vector Pulse Width Modulation

SVPWM technique is used as the modulation scheme in this thesis work. Therefore, it is described in detail in this section.

In the SVPWM technique, three-phase voltage references are represented as a space vector in the complex plane. Fundamental output voltage generated by SVPWM is 15.5% higher than that of SPWM technique. In addition, it gives less harmonic distortion of the load current, lower torque ripple in AC motors, and lower switching losses than SPWM [3]. Therefore, the SVPWM technique is widely used in many three-phase inverter applications.

1.3.2.1 Principle of Space Vector Modulation

The circuit model of a three-phase VSI is shown in Fig.1.3.2.1. The six power switches (S_1 to S_6) are controlled by the switching variables a , a', b , b', c and c'.

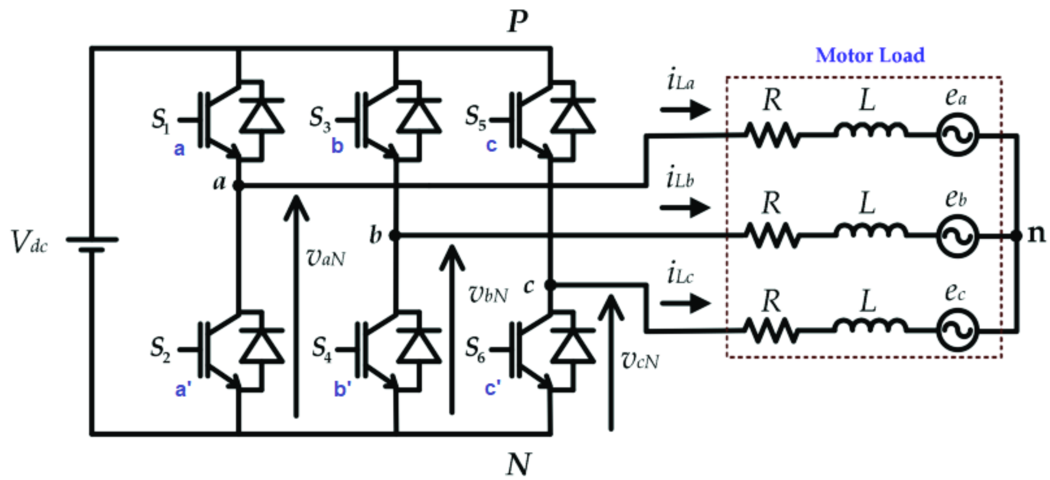

Figure 1.3.2.1: Three-phase Voltage Source PWM Inverter

Table 1.3.2.1. depicts the eight possible combinations of ON and OFF patterns for the three upper power switches. Switching states of the lower switches are complementary to the upper ones. The eight vectors are called the basic space vectors.

Table 1.3.2.1: Switching Vectors for SVPWM

Voltage Vectors	v_0	v_1	v_2	v_3	v_4	v_5	v_6	v_7
a	0	1	1	0	0	0	1	1
b	0	0	1	1	1	0	0	1
c	0	0	0	0	1	1	1	1

As the three-phase voltage references vary with time, the voltage reference vector rotates in the counter-clockwise direction in the complex plane as illustrated in Fig.1.3.2.2. This vector completes one revolution per electrical period of the reference voltage.

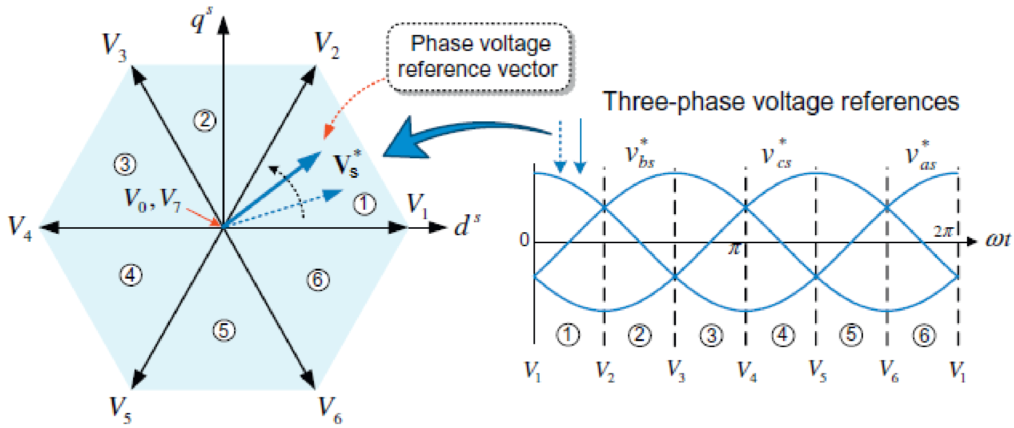

Figure 1.3.2.2: Space Voltage Vector Representation in Six Sectors of a Hexagon [3]

An arbitrary space vector can be represented by the sum of the two base vectors, between which the space vector is located, as illustrated in Fig.1.3.2.3.

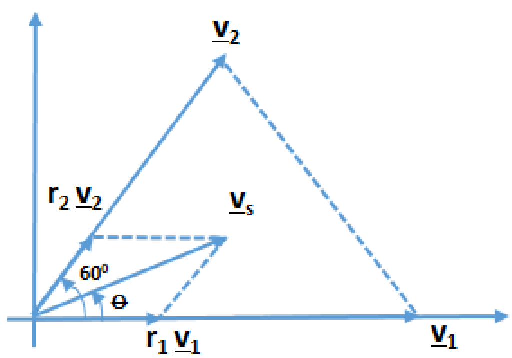

Figure 1.3.2.3: Voltage Space Vector Representation using Base Vectors

The reference voltage space vector $\underline{\mathrm{v}}_{\mathrm{s}}$ can be represented using base vectors $\underline{\mathrm{v}}_1$ and $\underline{\mathrm{v}}_2$ as;

$$\underline{v}_s = r_1.\underline{v}_1 + r_2.\underline{v}_2 \tag{1.3.2.1}$$

where r_1 and r_2 are coefficients.

Using the basic trigonometric relations;

$$r_1.\underline{v}_1 = \frac{2}{\sqrt{3}}.\underline{v}_s \sin(60 - \theta)$$

$$r_2.\underline{v}_2 = \frac{2}{\sqrt{3}}.\underline{v}_s \sin(\theta)$$

$\underline{v}_1$ and $\underline{v}_2$ are the length of the two vectors with the value of 2/3 V_{dc}.

By substituting $\underline{v}_1$ and $\underline{v}_2$ values r_1 and r_2 can be obtained as;

$$r_1 = \sqrt{3}\frac{v_s}{v_{dc}} \sin(60 - \theta) \tag{1.3.2.2}$$

$$r_2 = \sqrt{3}\frac{v_s}{v_{dc}} \sin(\theta) \tag{1.3.2.3}$$

For θ between 0^0 and 60^0, r_1 and r_1 are within the range [0,1]. They have the maximal value 1 when $\underline{v}_s$ coincides with $\underline{v}_1$ or $\underline{v}_2$.

The switching sequence is realized by timely activating the two vectors combined with zero vectors sequentially as shown in Fig.1.3.2.4. If the switching frequency is high, the approximation can precisely represent the reference vector.

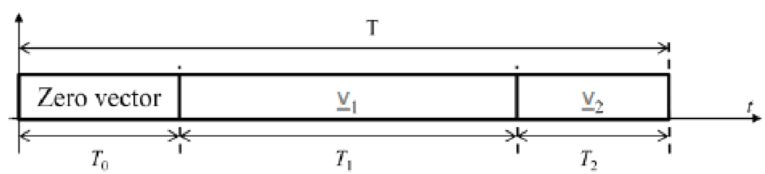

Figure 1.3.2.4: Combination of Vectors using Time Division

The three time durations are defined as;

$$\begin{cases} T_0 = (1 - r_1 - r_2)T \\ T_1 = r_1.T \\ T_2 = r_2.T \end{cases} \tag{1.3.2.4}$$

T is the PWM switching period, T_0 is the duration of zero vector, and T_1 and T_2 are the time durations of vector $\underline{u}_1$ and $\underline{u}_2$, respectively.

There are four possible switching sequences that can be used to generate the reference voltage vector. They are, asymmetric pulsation using v_0 as zero vector, symmetric pulsation using v_0 as zero vector, symmetric pulsation using v_7 as zero vector, and symmetric pulsation using both zeros vectors. Due to the generation of more harmonic noises in the first three methods, symmetric pulsation using both zeros vectors, shown in Fig.1.3.2.5 is most commonly used. [6]

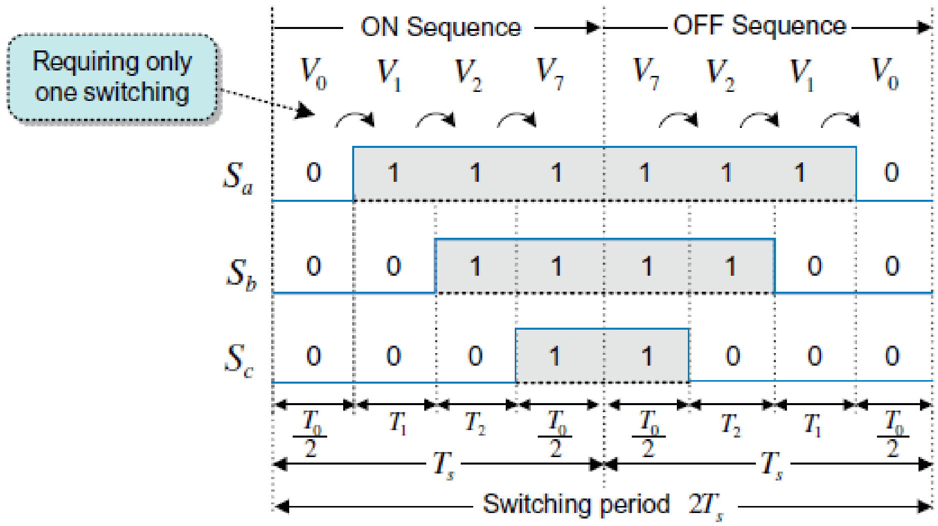

Figure 1.3.2.5: Switching Pulse Pattern for Three Phases in Sector 1 [3]

The shape of space vector modulated three-phase voltage waveforms are shown in Fig.1.3.2.6.

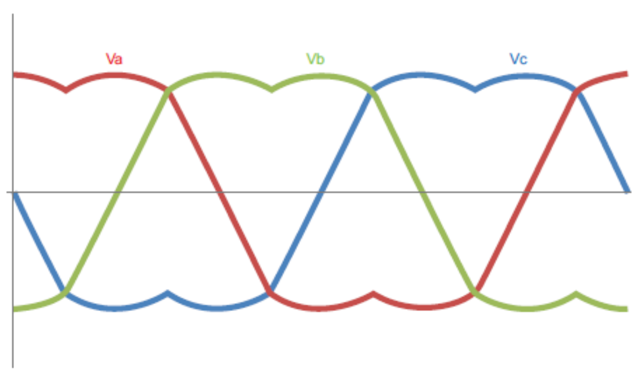

Figure 1.3.2.6: Space Vector Modulated Three Phase Voltage Waveforms

1.4 Vector Control of PMSM

In a PMSM, various parameters are coupled with each other, which makes the motor control strongly nonlinear. The basic idea of vector control is to achieve decoupling control of stator current by decoupling it into two-phase d-q axis currents; field producing current (i_d) and torque producing current (i_q). This allows to control of flux and torque of the motor independently. Vector control makes the model of PMSM linear [7]. A diagram of the vector control system implemented for controlling the speed of a PMSM in this thesis work is shown in Fig.1.4.1.

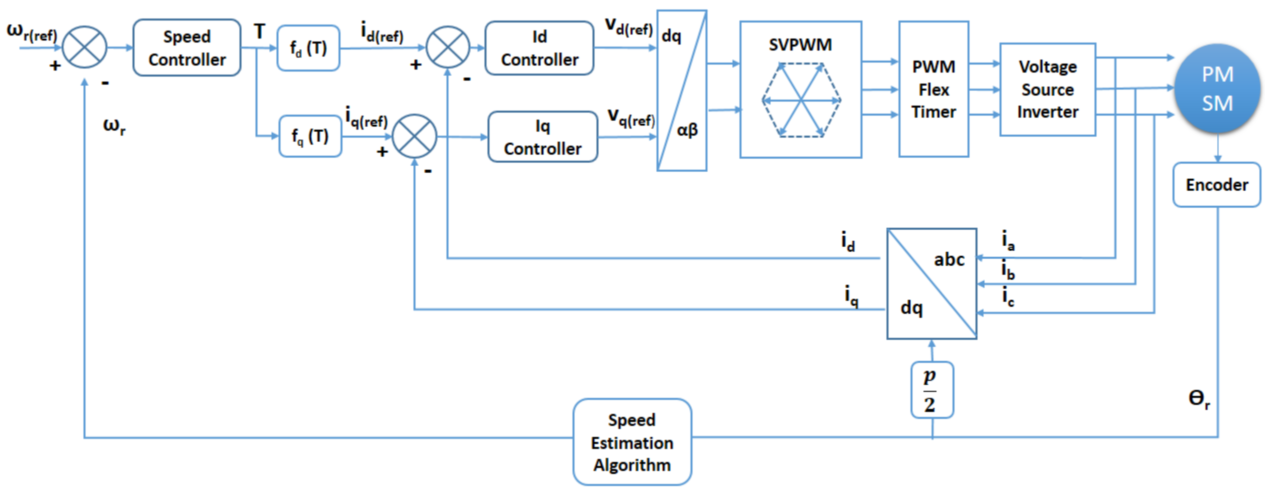

Figure 1.4.1: Vector Control System for Speed Controlling of an IPMSM

The measured three-phase currents are transformed into dq axis reference frame components. The angle of the rotor which is required for reference frame transformation is determined from encoder position information. Speed controller outputs torque reference from which current references for the current controller are obtained. Current controller generates reference voltages for the inverter. Using SVPWM, gate drive signals for inverter switches are generated as explained in section 1.3.2. All the other components of the diagram are analyzed and described in detail in subsequent chapters.

Chapter 2

Theoretical Modelling of PMSM

2.1 Reference Frame Transformation

In the synchronous machine model, magnetizing inductances are present which are position dependent. This position-dependent inductances occur due to electric circuits in relative motion and electric circuits with varying magnetic reluctance [8]. These time-dependent variables, increase the complexity of the machine model. Reference frame transformation is a concept used to eliminate all rotor position-dependent inductances from the voltage equations of the synchronous machine, making the machine model less complex and easy to control.

The transformation refers machine variables to a frame of reference which rotates at an arbitrary angular velocity. The three-phase abc variables are transformed into dq components which are orthogonal to each other. Several transformation techniques are introduced depending on the assignment of the speed of rotation of the reference frame. For this thesis work, synchronous reference frame which rotates in synchronism with the fundamental angular velocity of the stator variables is used.

Dq axis orientation of the synchronous reference frame can be selected in different ways. In our case, it is selected such that d-axis component is aligned 90 degrees behind a-axis, i.e at $t = 0$, the q-axis is aligned with phase a-axis as shown in Fig.2.1.1.

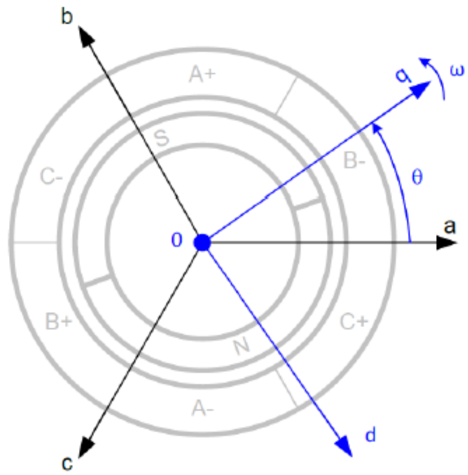

Figure 2.1.1: D and Q Axis Orientation

The relationship between angular position θ and velocity of the synchronous reference frame ω is given in,

$$\theta(t) = \int_0^t \omega(\tau)d\tau + \theta(0) \tag{2.1.1}$$

$\theta(0)$ is the angle between q axis and phase a-axis at t=0.

The transformation from abc to dq synchronous reference frame is obtained using transformation matrix K.

$$f_{qd0} = K * f_{abc} \tag{2.1.2}$$

$$f_{qd0} = \begin{bmatrix} f_q \\ f_d \\ f_0 \end{bmatrix}$$

$$f_{abc} = \begin{bmatrix} f_a \\ f_b \\ f_c \end{bmatrix}$$

$$K = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{2.1.3}$$

The total instantaneous power of a three-phase system can be expressed in abc variables as,

$$P_{abc} = v_a i_a + v_b i_b + v_c i_c \tag{2.1.4}$$

The total instantaneous power is the same regardless of the reference frame in which it is evaluated. Therefore, the total power can be expressed using qdo variables as;

$$P_{qd0} = P_{abc} = \frac{3}{2}(v_q i_q + v_d i_d + v_0 i_0) \tag{2.1.5}$$

The 3/2 factor comes due to the choice of the constant used in the transformation [3].

2.2 D-q axis Model of a Permanent Magnet Synchronous Motor

As motioned before, for reference frame transformation, the speed of dq axis reference frame is chosen as the synchronous speed of the motor. The direct axis is chosen along the permanent magnetic flux vector as shown in Fig.2.2.1. Angle θ is obtained using encoder readings which is mounted on rotor shaft.

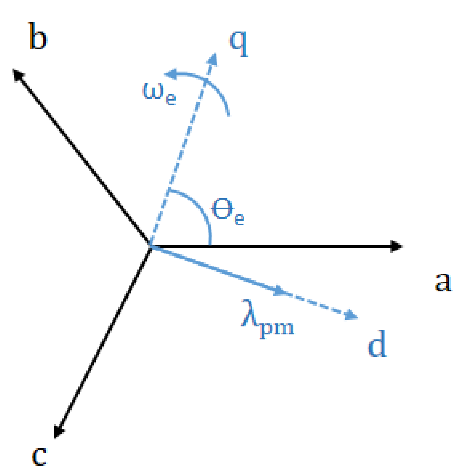

Figure 2.2.1: Alignment of d-axis with the Magnetic Flux Vector in a PMSM

2.2.1 Voltage and Flux Equations

In synchronous reference frame, the voltage equation can be expressed as,

$$v_{qd0} = R_s * i_{qdo} + \frac{d\lambda_{qd0}}{dt} + \omega_e * \lambda_{dq0} \tag{2.2.1.1}$$

λ_{qd0} can be obtained as;

$$\lambda_{qd0} = \begin{bmatrix} -\lambda_q \\ \lambda_d \\ 0 \end{bmatrix} \tag{2.2.1.2}$$

Flux equation is defined as,

$$\lambda_{qd0} = \lambda_{pm,qd0} + L_{qd0} i_{qd0} \tag{2.2.1.3}$$

λ_{pm} is the flux linkage caused by the permanent magnet. Dq reference frame is chosen such that d-axis is aligned with the rotor flux axis. Hence, permanent magnet flux along q-axis does not exist.

Eq.2.2.1.2 can be modified to result;

$$\lambda_{qd0} = \begin{bmatrix} -\lambda_q \\ \lambda_d \\ 0 \end{bmatrix} = \begin{bmatrix} -L_q i_q \\ \lambda_{pm} + L_d i_d \\ 0 \end{bmatrix} \tag{2.2.1.4}$$

Inductance matrix is given by;

$$L_{qd0} = \begin{bmatrix} L_q & 0 & 0 \\ 0 & L_d & 0 \\ 0 & 0 & L_0 \end{bmatrix} \tag{2.2.1.5}$$

L_d and L_q are d and q-axis stator self-inductances.

D-axis and q-axis voltage equations can then be derived in to;

$$v_q = R_s i_q + \frac{d(L_q i_q)}{dt} + \omega_e(\lambda_{pm} + L_d i_d) \tag{2.2.1.6}$$

$$v_d = R_s i_d + \frac{d(L_d i_d)}{dt} - \omega_e (L_q i_q) \tag{2.2.1.7}$$

Here, R_s is stator resistance per-phase. ω_e is electrical angular velocity and θ_e is electrical angle.

2.2.2 Torque Equation

The power equation can be used to derive the torque equation in synchronous reference frame. The input power is expressed in the dq axes as;

$$P_{in} = \frac{3}{2}(v_q i_q + v_d i_d) \tag{2.2.2.1}$$

By substituting d and q axis voltages and flux linkages into Eq.(2.2.2.1),

$$\begin{aligned} P_{in} =& \frac{3}{2}[(R_s i_q + \frac{d\lambda_q}{dt} + \omega_e \lambda_d) i_q + (R_s i_d + \frac{d\lambda_d}{dt} - \omega_e \lambda_q) i_d] \\ =& \frac{3}{2}[R_s(i_q^2 + i_d^2) + i_q \frac{d\lambda_q}{dt} + i_d \frac{d\lambda_d}{dt} + \omega_e \lambda_d i_q - \omega_e \lambda_q i_d] \end{aligned}$$

The first term represents the stator copper losses and the second two terms represent the variation in the magnetic energy. Hence, the last two terms represent the mechanical output power that develops the output torque. [3]

$$P_{out} = \frac{3}{2}\omega_e[\lambda_d i_q - \lambda_q i_d] \tag{2.2.2.2}$$

By substituting λ_d and λ_q in to Eq.2.2.2.2,

$$P_{out} = \frac{3}{2}\omega_e[\lambda_{pm} i_q + (L_d - L_q) i_d i_q] \tag{2.2.2.3}$$

The relationship between the output power and output torque is given in;

$$P_{out} = T.\omega_r \tag{2.2.2.4}$$

Electrical speed and the mechanical speed of the motor is related as,

$$\omega_e = P.\omega_r \tag{2.2.2.5}$$

where P is the number of pole pairs.

Using above relationships, Eq.2.2.2.3 can be modified to obtain the torque equation as;

$$T = \frac{3P}{2}[\lambda_{pm} i_q + (L_d - L_q) i_d i_q] \tag{2.2.2.6}$$

Chapter 3

Theoretical Analysis of Control Loops Design

3.1 Current Controller Design

For PMSMs, current control loop can be considered as the torque control loop, because torque is indirectly controlled by applying a current demand. The current loop is the inner loop in a servo control system, thus it should have a quicker response than outer loops.

In the next subsections two types of current regulator models are discussed.

3.1.1 Types of Current Regulators

3.1.1.1 Cross-Coupling Decoupling Synchronous Frame PI Current Regulator

D-axis and q-axis voltage equations in the synchronous reference frame are;

$$v_q = R_s i_q + \frac{d(L_q i_q)}{dt} + \omega_e \lambda_{pm} + \omega_e L_d i_d \tag{3.1.1.1}$$

$$v_d = R_s i_d + \frac{d(L_d i_d)}{dt} - \omega_e L_q i_q \tag{3.1.1.2}$$

There are speed-dependent terms in the synchronous reference frame expressions given in Eq.3.1.1.1 and Eq.3.1.1.2. The term $\omega_e \lambda_{pm}$ represents the back-emf voltage. The terms $\omega_e L_d i_d$ and $\omega_e L_q i_q$ voltages are cross-coupled between d- and q-axes. Thus d and q axis controls are influenced by each other, i.e. change in one may affect the other. The effect of these components may become significant at high operating frequencies, incurring an oscillatory current response. If the bandwidth of the current controller is large enough, their influence can be reduced. However, bandwidth is limited by switching frequency and current sampling frequency. Also, the system becomes more sensitive to noise when controller gains are high [3]. Therefore, to eliminate the effect of these disturbance components and improve the performance of the current control, the feed-forward compensation of these voltages should be employed. Feedforward terms decouple the d-q axis Proportional and Integral (PI) controllers and help make the current transient response similar to the designed response.

Fig.3.1.1.1 depicts a block diagram of the current regulator with added back emf compensation and cross-coupling decoupling terms.

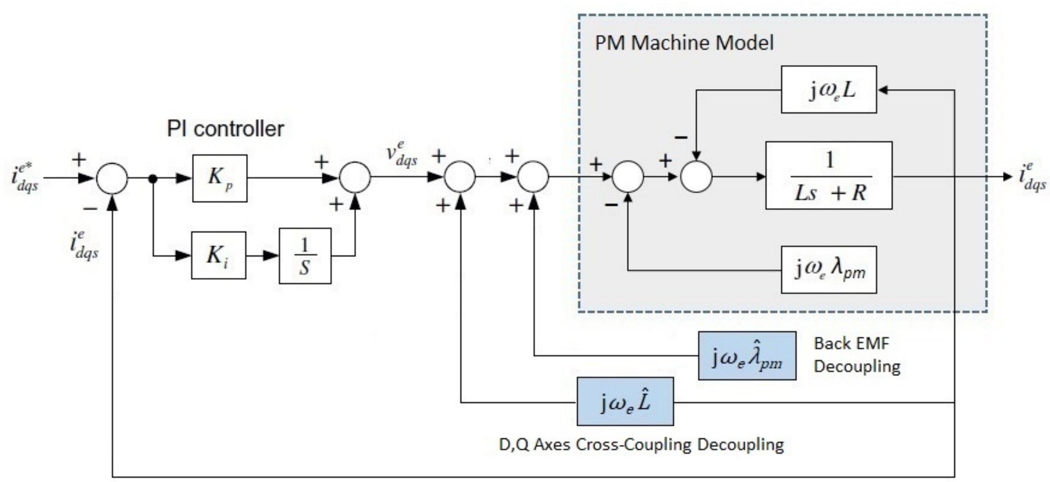

Figure 3.1.1.1: Cross-coupling Decoupling Synchronous Frame current Regulator

3.1.1.2 Complex Vector Synchronous Frame Current Regulator

Performance of the cross-coupling decoupling synchronous frame current regulator depends on the accuracy of the measured system parameters, i.e. d and q axis inductance values. If system parameter measurements are inaccurate, the estimated value of the feed-forward compensation becomes inaccurate. As a solution to this issue, complex vector synchronous frame current regulator was introduced. As shown in Fig.3.1.1.2, it has decoupling control inside the PI controller without using any system parameters. Therefore, the performance of the complex vector synchronous frame current regulator is less sensitive to errors in system parameters and is independent of the operating frequency.

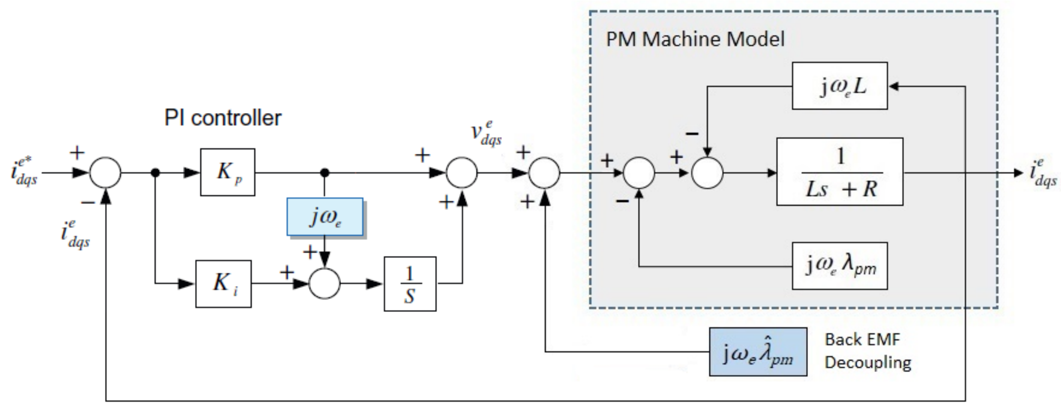

Figure 3.1.1.2: Complex Vector Synchronous Frame Current Regulator

3.1.2 Current Controller Tuning

The first generation of electric servo drives, which used separately excited dc motors, had a completely analog control system. Thus, root locus techniques in continuous domain (S domain) was used to design the current control loop. Nowadays, microprocessors are being used and the control is implemented in digital domain. Therefore, it is necessary to design the current loop in the Z-domain. However, If switching frequency is high, effect of sampling can be ignored and analysis can be done in the continuous time domain.

In the next sub-sections current controller analysis in both S-domain and Z-domain are discussed. For the design, VSI is considered as an ideal, unity gain model which generates only the fundamental component of the reference voltage. Furthermore, switching losses and switching harmonics are neglected.

3.1.2.1 S-domain Analysis

When designing the current controller, the main considerations are short response time and zero steady-state error. PI type controller is used for this purpose. Controller gains are determined considering the plant model and the bandwidth of the loop. S-domain current control loop model with RL load is shown in Fig.3.1.2.1.

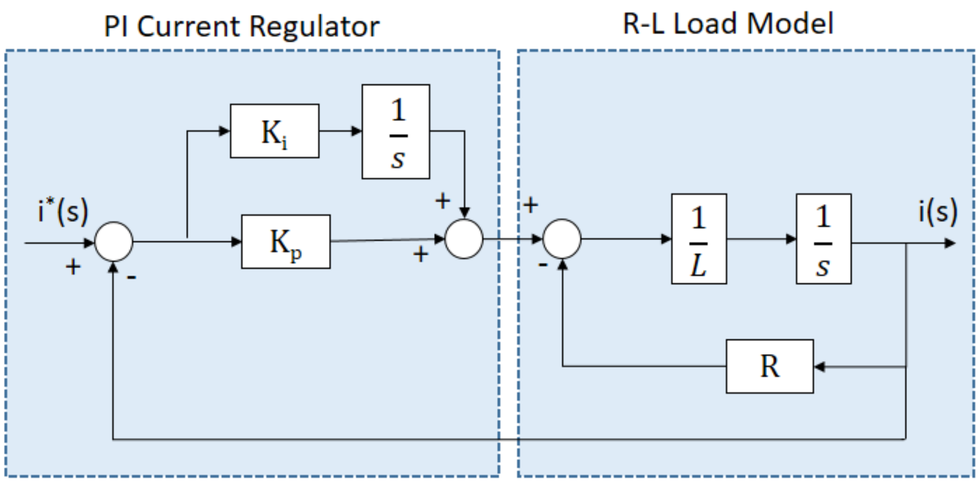

Figure 3.1.2.1: PI Current Regulator with RL Load Model in S-domain

Controller transfer function C(s) and plant transfer function P(s) can be expressed as;

$$C(s) = K_p + \frac{K_i}{s} = K_p\left[\frac{s + K_i/K_p}{s}\right] \tag{3.1.2.1}$$

$$P(s) = \frac{1}{sL + R} = \frac{1}{L(s + R/L)} \tag{3.1.2.2}$$

Open Loop Transfer Function (OLTF) can then be obtained as;

$$\begin{aligned} OLTF(s) =& C(s) * P(s) \\ =& \frac{K_p}{L}\,\frac{(s + K_i/K_p)}{s(s + R/L)} \end{aligned} \tag{3.1.2.3}$$

Controller zero is chosen such that the plant pole is cancelled with the controller zero.

$$\frac{K_i}{K_p} = \frac{R}{L} \tag{3.1.2.4}$$

Current controller OLTF can now be simplified in to;

$$OLTF(s) = \frac{K_p}{sL} \tag{3.1.2.5}$$

Close Loop Transfer Function (CLTF) can be obtained using the OLTF as;

$$\begin{aligned} CLTF(s) =& \frac{OLTF(s)}{1 + OLTF(s)} \\ =& \frac{\frac{K_p}{sL}}{1 + \frac{K_p}{sL}} = \frac{\frac{K_p}{L}}{s + \frac{K_P}{L}} \end{aligned} \tag{3.1.2.6}$$

Close loop pole is located at -K_p/L. The relationship between the desired bandwidth of the controller f_{BW} and the close loop pole can be expressed as;

$$2 * \pi * f_{BW} = \frac{K_p}{L} \tag{3.1.2.7}$$

From Eq.3.1.2.4 and Eq.3.1.2.7, K_p and K_i can be calculated as;

$$K_p = 2 * \pi * f_{bw} * L \tag{3.1.2.8}$$

$$K_i = 2 * \pi * f_{bw} * R \tag{3.1.2.9}$$

The bandwidth of a current controller is limited by the switching frequency of the power electronic converter and the Analog to Digital Converter (ADC) sampling period. If the current is sampled once every switching period, the maximum available bandwidth can be up to 1/20 of the switching frequency. [3]

S-domain root locus plot for a 300Hz current control loop bandwidth is depicted in Fig.3.1.2.2.

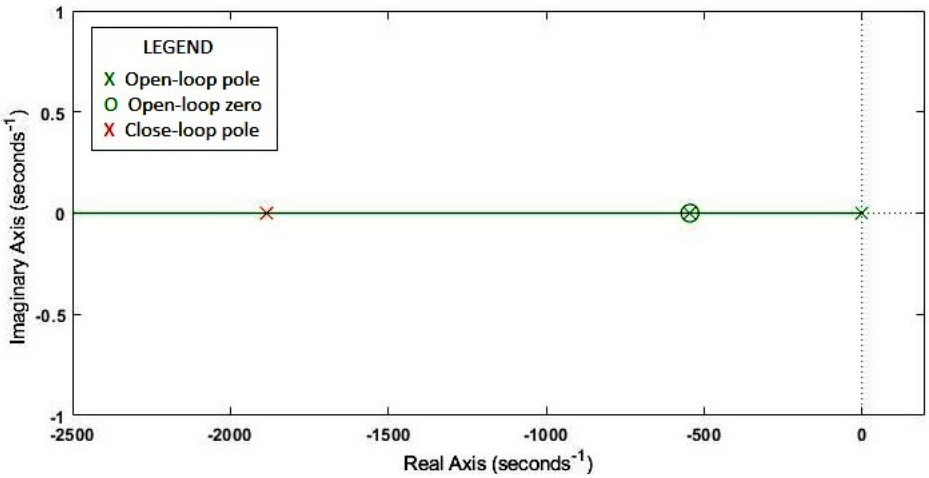

Figure 3.1.2.2: Root Locus Plot with Zero-pole Cancellation in S-domain at 300Hz Bandwidth

3.1.2.2 Z-domain Analysis

The actual physical system runs in continuous time domain while the controller gives signal in discrete-time domain due to the sampling in ADC. Therefore, for an accurate representation of the signals in Z-domain, the ADC is preceded by a Sample and Hold/Zero Order Hold (ZOH) block. Fig.3.1.2.3. shows the zero-order-hold discretization of a continuous time linear model.

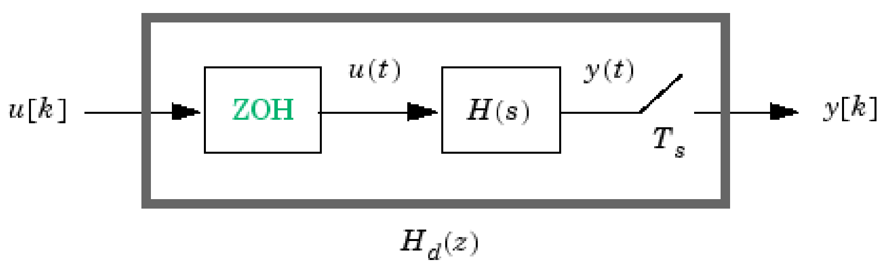

Figure 3.1.2.3: ZOH Discretization of a Continuous Time Model

The transfer function H(z) in discrete domain by considering the effect of ZOH is given by;

$$H(z) = (1 - z^{-1}) \quad \boldsymbol{Z}\left\{\frac{1}{s}.H(s)\right\} \tag{3.1.2.10}$$

Using Eq.3.1.2.10. Z-domain controller transfer function C(z) and plant transfer function P(z) can be derived in to;

$$C(z) = K_p + K_i T_s \Big(\frac{z}{z-1} \Big) \tag{3.1.2.11}$$

$$P(z) = \frac{(1 - e^{-T_s/\tau}) z^{-1}}{R(1 - e^{-T_s/\tau} z^{-1})} \tag{3.1.2.12}$$

Here, T_s is the sample time and τ = R/L is the electric time constant of the motor.

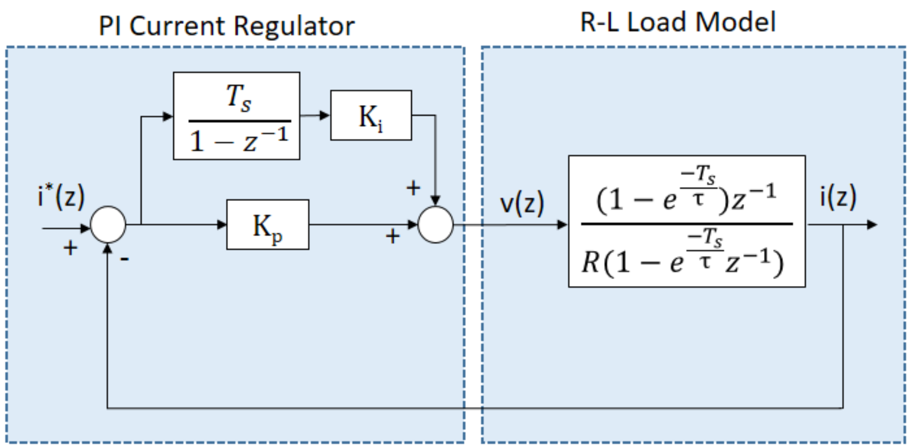

Figure 3.1.2.4: PI Current Regulator with RL Load Model in Z-domain

Figure 3.1.2.4. shows the Z-domain model of the current control loop. Z-domain open-loop transfer function of the system OLTF(z), can be obtained as;

$$OLTF(z) = \frac{K(z - \delta_c)}{(z-1)(z - e^{-T_s/\tau})} \tag{3.1.2.13}$$

Where the gain K and the controller zero δ_c are defined as:

$$K = \frac{(K_p + K_i T_s)}{R}(1 - e^{-T_s/\tau}) \tag{3.1.2.14}$$

$$\delta_c = \frac{K_p}{K_p + K_i T_s} \tag{3.1.2.15}$$

For pole-zero cancellation, the controller zero δ_c is chosen to be equal to the machine open-loop pole.

$$\delta_c = e^{-T_s/\tau}$$

Close loop transfer function CLTF(z) can then be obtained as;

$$CLTF(z) = \frac{K}{z - (1 - K)} \tag{3.1.2.16}$$

Here, (1-K) is the close loop pole. In order to approximately achieve the desired closed-loop bandwidth f_{BW}, the closed-loop pole p_{cl} is placed at:

$$p_{cl} = e^{-2\pi f_{BW} T_s} \tag{3.1.2.17}$$

Using the above relationships, it can be shown that in order to achieve both zero-pole cancellation and the desired closed loop pole placement, K_i and K_p must be selected as follows;

$$K_i = \frac{R}{T_s}(1 - p_{cl}) \tag{3.1.2.18}$$

$$K_p = \frac{R(1 - p_{cl})e^{-T_s/\tau}}{(1 - e^{-T_s/\tau})} \tag{3.1.2.19}$$

Z-domain Root locus plot with zero-pole cancellation, for a 300Hz current control loop bandwidth is shown in Fig.3.1.2.5.

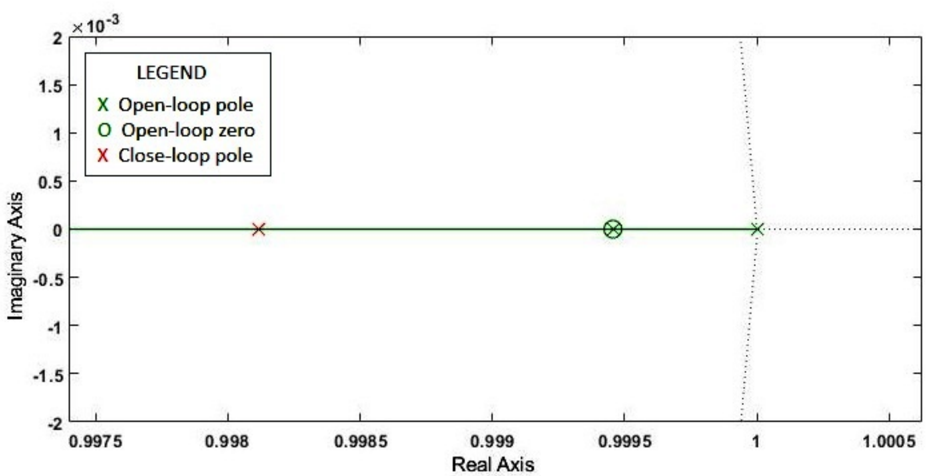

Figure 3.1.2.5: Root Locus Plot with Zero-pole Cancellation in Z-domain

3.2 Integrator Anti-wind up

The voltage reference applied to a motor which is provided by the PI controller output is limited by DC-link voltage and limitations imposed by inverter and motor design. When the voltage hits its limit, the output of the controller will be saturated. But the integrator will continue to integrate the error, so the integrator output can reach a large value. This is known as integral windup. As a result of this, system response can have a large overshoot and a long setting time [3]. In order to prevent integrator windup, an anti-windup algorithm should be implemented within the control structure so that the accumulated value of the integrator is kept at a proper value when the saturation occurs.

The anti-windup algorithm using back-calculation method shown in Fig.3.2.1 is widely used because of its satisfactory dynamic performance. In this scheme, the anti-windup path doesn't work as long as the voltage is not saturated, when it does, input to the integral part of the PI regulator will be reduced by a gain of K_a.

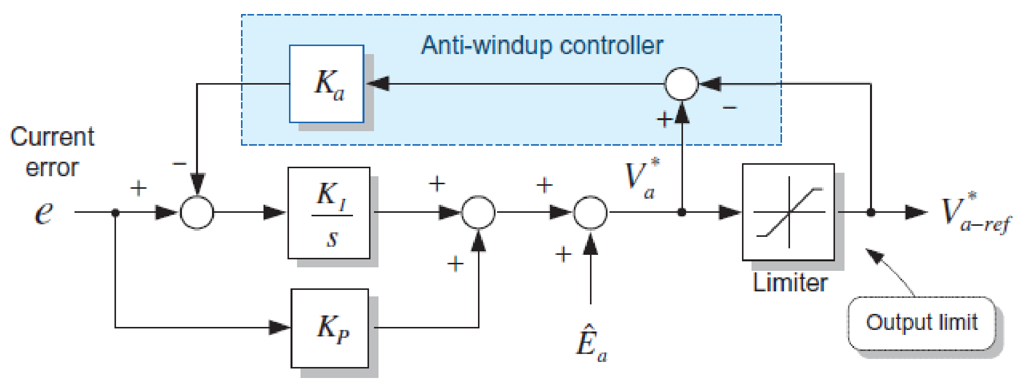

Figure 3.2.1: Anti-windup Algorithm Using Back Calculation Method

The gain K_a is given by,

$$K_a = \frac{1}{K_p} \tag{3.2.1}$$

3.3 Dead Time Compensation

When both switches in an inverter leg are turned on at the same time, the DC source is short-circuited, resulting in a large current that can destroy switches. Thus, it is essential to insert a time delay in control signals to ensure that the switching states of the two switches change in a complementary manner. This time delay is called dead time or blanking time which is applied prior to the gate signal of the turning-on device. The blanking time depends on the type and the size of a semiconductor device. A value of 1-3 μs is typical for a medium-power IGBT [3].

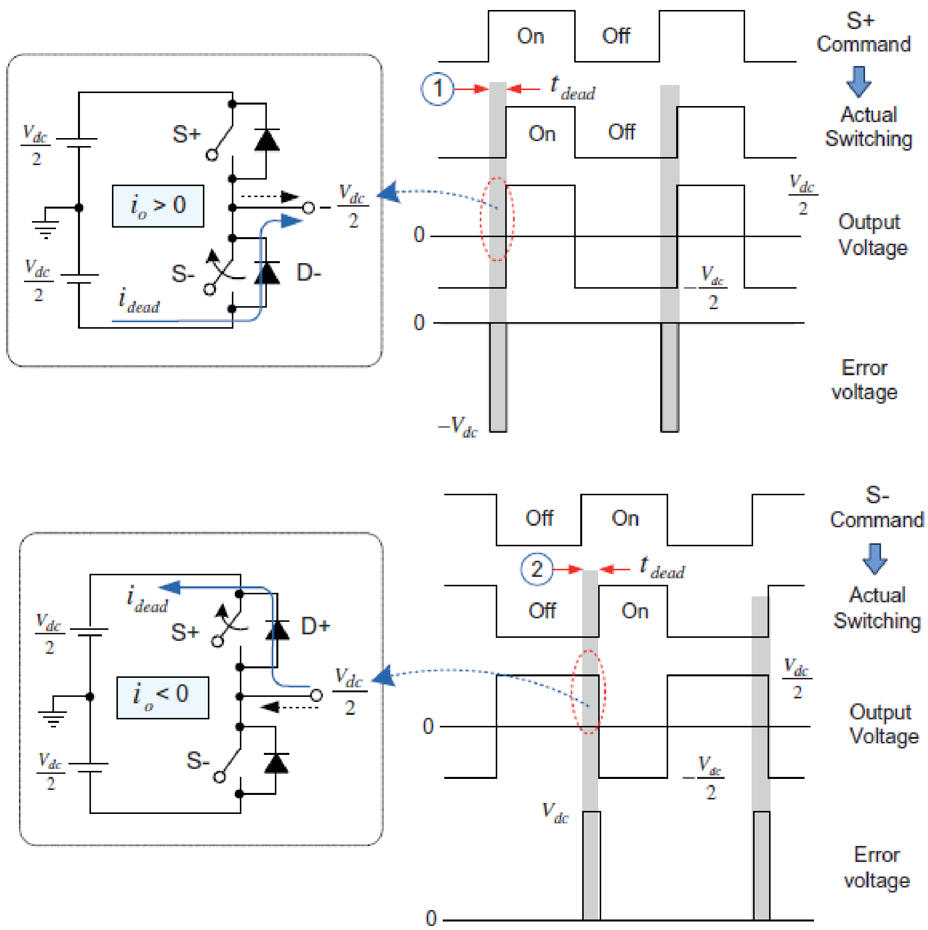

Figure 3.3.1: Effect of Dead-Time on the Output Voltage [3]

During the dead time, both the switches are turned OFF, so the current conducts through freewheeling diode D+ or D-. Which one of the diodes will conduct depends on the direction of the current flow. As shown in Fig.3.3.1, when the current is positive (flows towards the load) diode D- conducts and negative voltage will appear at the output terminal. On the other hand, when current is negative, diode D+ conducts and output terminal voltage will be positive. Therefore, during the dead time, the output voltage of the inverter can be different from the voltage command. This causes fundamental voltage drop. Moreover, undesirable harmonic components can appear in the output voltage, which causes overall distortion of the inverter output voltage and current waveforms [9]. Effect of dead time increases with switching frequency. Thus it is necessary to compensate for the voltage distortion caused by the dead-time effect.

A dead time compensation technique was introduced in [10]. In this strategy the instantaneous sector (1-6) of the current space vector is determined and compensating voltage signals are calculated in dq synchronous reference frame as;

$$V_d^* = \frac{4}{3} V_f \cos\left[\frac{\pi}{3} Trunc\Big(\frac{\omega_e t + \theta^* + \frac{\pi}{6}}{\frac{\pi}{3}}\Big) - \omega_e t\right] \tag{3.3.1}$$

$$V_q^* = \frac{4}{3} V_f \sin\left[\frac{\pi}{3} Trunc\Big(\frac{\omega_e t + \theta^* + \frac{\pi}{6}}{\frac{\pi}{3}}\Big) - \omega_e t\right] \tag{3.3.2}$$

where $V_f = V_{dc}(T_{dt}/T_s)$ and $\theta^* = \tan^{-1}(i_q^*/i_d^*)$ and ω_e is electrical angular frequency of the synchronous reference frame.

As mentioned in section 2.1, dq axis orientation of the synchronous reference frame can be selected in different ways. These equations are derived for the most commonly used orientation in which d axis component is aligned with phase A axis at t=0. Therefore, these equations should be manipulated according to the selected dq axis orientation.

3.4 Vector Tracking Speed Observer

Estimation of speed is necessary for the closed-loop speed control, back-emf compensation, and dq axis decoupling. Rotor position obtained from the encoder signals can be used to estimate the speed of the rotor. Vector Tracking Observer (VTO) uses vector cross-product phase detection to estimate the speed from the rotor position. A block diagram of the VTO model is shown in Fig.3.4.1. A Proportional, Integral and Derivative (PID) regulator is used in this design.

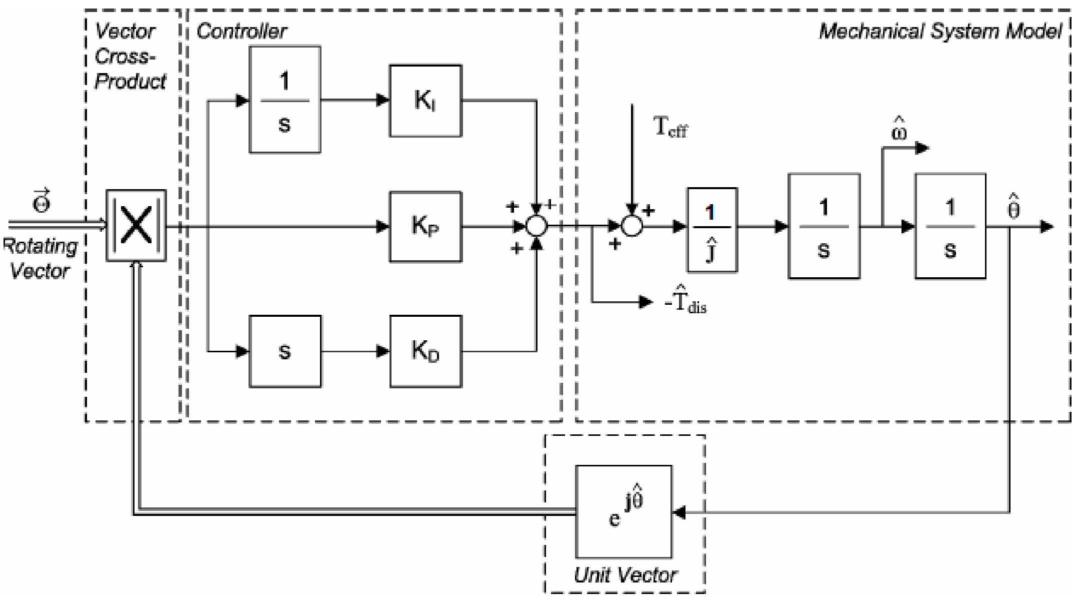

Figure 3.4.1: Structure of the VTO in S-domain [11]

It should be noted that the observer model shown in Fig.3.4.1 estimates the mechanical speed of the motor and input to the observer is the mechanical angle of the rotor.

Mechanical model of the system is given by,

$$T - T_L = J\frac{d\omega_m}{dt} + B\omega_m \tag{3.4.1}$$

where ω_m is the mechanical speed of the rotor, T and T_L are applied electromagnetic torque and load torque of the machine respectively. J is the total inertia of the machine. Friction B is negligible thus it is ignored to simplify the model.

VTO takes the vector cross-product between angle θ read through encoder, and position from the observer output $\hat{\theta}$, to determine the error. If the difference between

θ and $\hat{\theta}$ is small, the following relationship is valid;

$$Error = \sin(\theta).\cos(\hat{\theta}) - \cos(\theta).\sin(\hat{\theta}) = \sin(\theta - \hat{\theta}) \simeq \Delta\theta \tag{3.4.2}$$

If the position estimation error exceeds 90^0, the previous assumption is no longer valid, and the system will attain the unstable region as shown in the Fig.3.4.2.

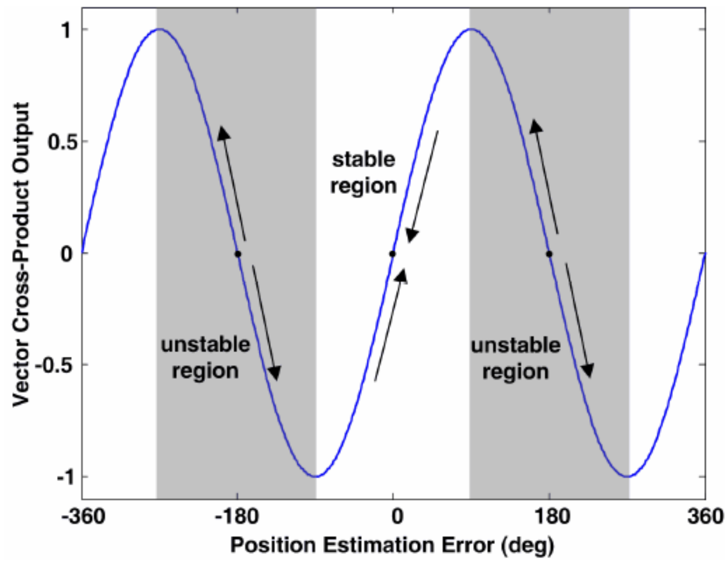

Figure 3.4.2: Vector Cross-Product Output as a Function of the Estimation Error [11]

However, local stability guarantees that the system will convergence towards the nearest point where the vector cross-product output is zero. This occurs when,

$$\Delta\theta = \theta - \hat{\theta} = k\pi;\ k = 0, \pm1, \pm2, \pm3, ...$$

In the alternative structure shown in Fig.3.4.3, un-realizable derivative term is avoided by moving it to the velocity state.

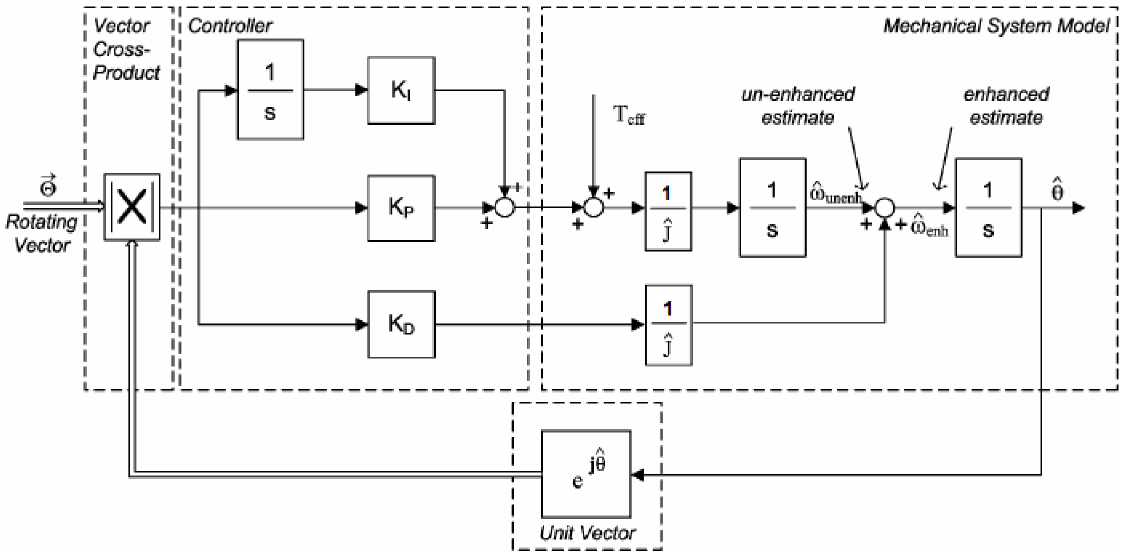

Figure 3.4.3: Alternative Structure of a VTO [11]

In this model, there are two speed estimates, "un-enhanced" estimate directly after the state integrator and "enhanced" estimate after adding the derivative term. Both estimates can be used. The enhanced signal is the same as the speed estimate shown in Fig.3.4.1. However, it will be noisier than the un-enhanced speed estimate because of the contribution of the controller's derivative action.

3.4.1 Vector Tracking Observer Tuning in Z-domain

The tuning of the VTO is done in Z-domain since the system works in discrete-time domain. Fig.3.4.1.1 shows a block diagram of the VTO modelled in Z-domain. The desired dynamic response can be achieved by proper selection of the three controller gains K_P,K_I and K_D.

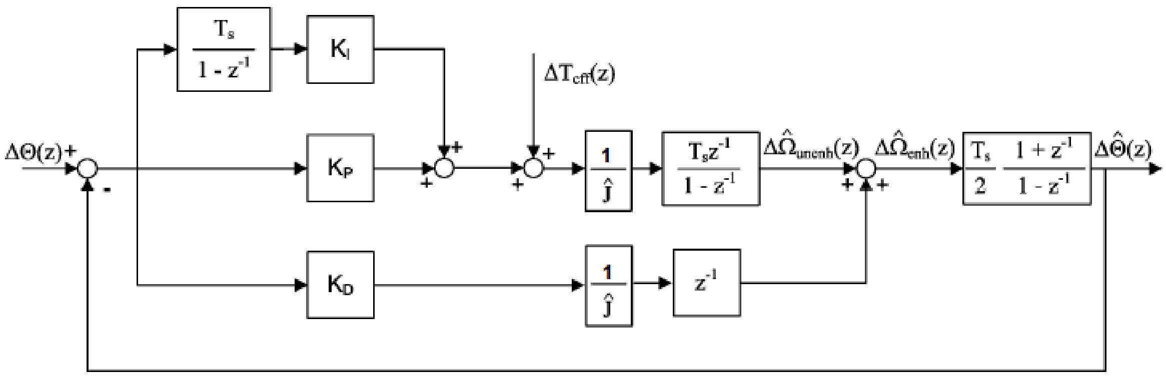

Figure 3.4.1.1: Z-Domain Model of the VTO [11]

In order to simplify calculations, four new gains K_1,K_2,K_3 and K_{eff} are introduced by combining the system parameters and gains accordingly. The modified model is shown in Fig.3.4.1.2.

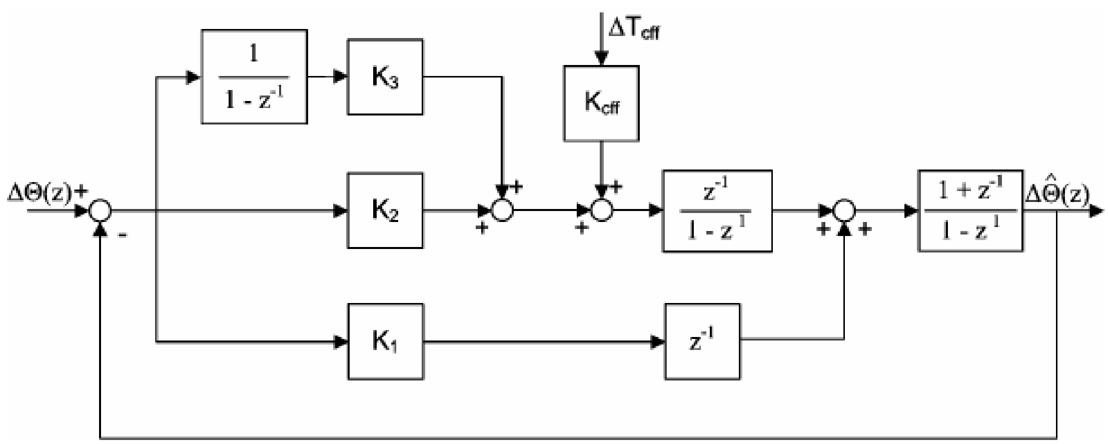

Figure 3.4.1.2: VTO Model with Lumped Parameters and Gains [11]

The new gains are given by,

$$\begin{cases} K_1 = \frac{K_D T_s}{2\hat{j}} \\ K_2 = \frac{K_P T_s^2}{2\hat{j}} \\ K_3 = \frac{K_I T_s^3}{2\hat{j}} \\ K_{eff} = \frac{T_s^2}{2\hat{j}} \end{cases} \tag{3.4.1.1}$$

Feed-forward signal can be excluded in the model as it doesn't have any influence on the tuning procedure. Furthermore, the parallel paths in the observer model can be algebraically manipulated in order to clearly see the three cascaded loops as shown in Fig.3.4.1.3.

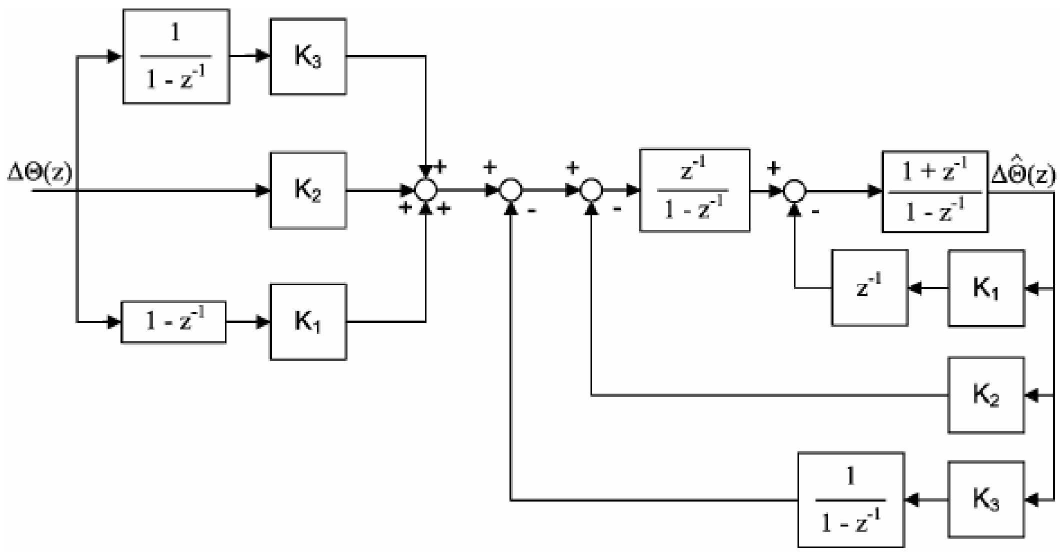

Figure 3.4.1.3: Equivalent Observer Model With Control Loops Explicitly Shown [11]

The transfer function of the derivative loop is given by,

$$\frac{z^2 + z}{z^2 + (K_1 - 1)z + K_1} \tag{3.4.1.2}$$

and the characteristics equation is,

$$z^2 + (K_1 - 1)z + K_1 = 0 \tag{3.4.1.3}$$

The pole z_1 can be placed according to the desired bandwidth of the derivative loop and the other pole is free to vary.

$$z_1 = e^{\frac{-T_s}{\tau_1}} \tag{3.4.1.4}$$

where τ_1 is the time constant of the derivative loop. The derivative loop is tuned to have the highest bandwidth.

K_1 can be obtained from the characteristics equation as;

$$K_1 = z_1 \frac{1 - z_1}{1 + z_1} \tag{3.4.1.5}$$

K_1 should be selected so that complex poles are avoided. From the root locus plot shown in Fig.3.4.1.4, the maximum limit for K_1 is found to be 0.1716, which corresponds to having two coexistent poles at z = 0.4142.

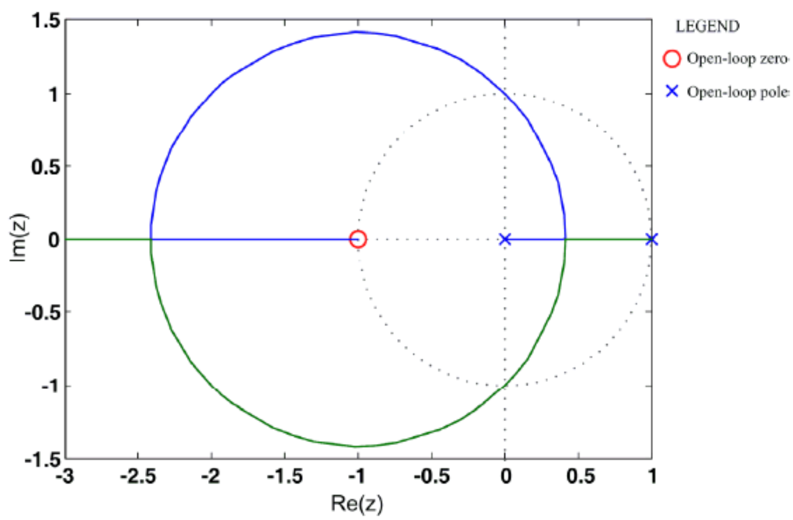

Figure 3.4.1.4: Root Locus Plot for the Derivative Loop

Once the derivative loop is tuned, tuning of the proportional loop can be analyzed. The overall transfer function of the combined proportional and derivative loops is:

$$\frac{z^2 + z}{z^3 + (K_1 + K_2 - 2)z^2 + (K_2 + 1)z - K_1} \tag{3.4.1.6}$$

and the characteristic equation is;

$$z^3 + (K_1 + K_2 - 2)z^2 + (K_2 + 1)z - K_1 = 0 \tag{3.4.1.7}$$

The proportional loop pole z_2 is set to;

$$z_2 = e^{\frac{-T_s}{\tau_2}} \tag{3.4.1.8}$$

where τ_2 is the desired time constant of the proportional loop. K_2 obtained from the characteristic equation is:

$$K_2 = \frac{-z_2^3 + (2 - K_1)z_2^2 - z_2 + K_1}{z_2(z_2 + 1)} \tag{3.4.1.9}$$

K_2 also has its maximum limit to avoid complex poles and it depends on the value of K_1. The root locus for the derivative and proportional loops is shown in Fig.3.4.1.5 for a particular K_1 value.

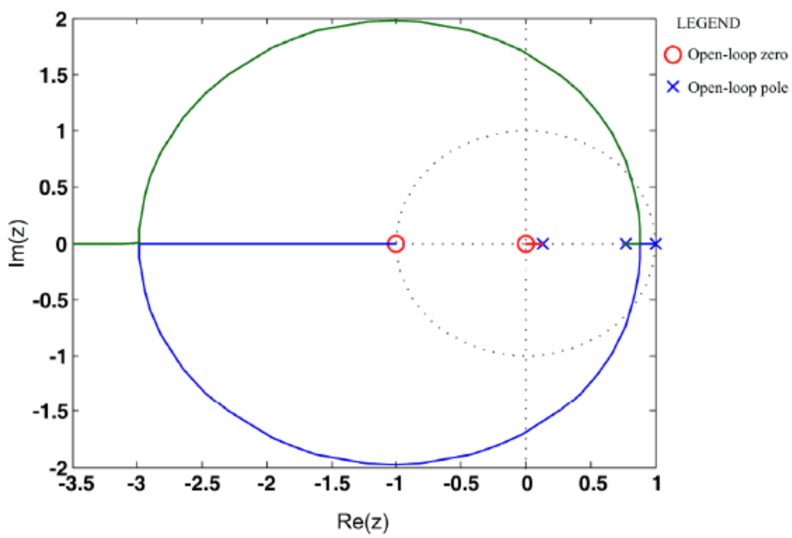

Figure 3.4.1.5: Root Locus Plot for the Derivative and Proportional Loop

Finally, in order to tune the integral loop, transfer function of the complete system is considered which is given in;

$$\frac{z^3 + z^2}{z^4 + (K_1 + K_2 + K_3 - 3)z^3 + (K_3 - K_1 + 3)z^2 - (K_1 + K_2 + 1)z + K_1} \tag{3.4.1.10}$$

and the characteristic equation is:

$$z^4 + (K_1 + K_2 + K_3 - 3)z^3 + (K_3 - K_1 + 3)z^2 - (K_1 + K_2 + 1)z + K_1 = 0 \quad (3.4.1.11)$$

The integral loop pole z_3 is set to;

$$z_3 = e^{\frac{-T_s}{\tau_3}} \quad (3.4.1.12)$$

where τ_3 is the desired time constant of the integral loop. K_3 is obtained by:

$$K_3 = \frac{-z_3^4 - (K_1 + K_2 - 3)z_3^3 + (K_1 - 3)z_3^2 + (K_1 + K_2 + 1)z_3 - K_1}{z_3^3 + z_3^2} \quad (3.4.1.13)$$

K_3 limit depends on the choice of both K_1 and K_2. A zoomed view of an example root locus plot for the complete system is shown in Fig.3.4.1.6.

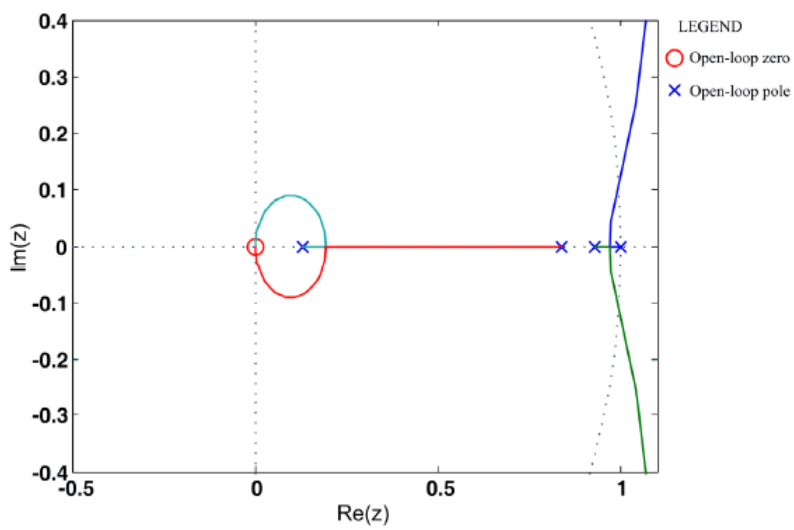

Figure 3.4.1.6: Root Locus Plot Root for the Derivative Proportional and Integral Loops

The time constant of the three control loops should be selected such that, $\tau_1 >> \tau_2 >> \tau_3$ and a factor of at least 10 between the time constants should be considered [11].

3.5 Speed Controller Design

The speed control loop is the outer loop, which generates torque references for the current regulator. A simple control scheme can be implemented in which d-axis current reference is zero and speed controller provides the q-axis current reference to produce torque. It will work well, but the motor will not operate at full potential as reluctance torque component is not utilized. Therefore, for salient pole machines, Maximum Torque per Ampere (MTPA) technique is preferred for speed control loop design.

The speed control loop is also designed together with the anti-windup algorithm which is explained in section 3.3. Either current reference or torque reference saturation can be considered for designing the anti-windup scheme.

3.5.1 Maximum Torque per Ampere (MTPA) Control

Torque produced by an IPMSM machine derived in section 2.2.2. as;

$$T = \frac{3P}{2}[\lambda_{pm} i_q + (L_d - L_q) i_d i_q] \tag{3.5.1.1}$$

In an IPMSM machine, d-axis inductance L_d and q-axis inductance L_q are not equal. The first term of the Eq.3.5.1.1 represents the magnetic torque, and the second term represents reluctance torque, which is produced by the difference between the inductance values of the two axes. To produce the reluctance torque, both the d and the q-axes components of the stator current are required. In most commonly used parallel topology of IPMSMs, $L_q > L_d$ [3]. Therefore, the d-axis current must have a negative polarity for the reluctance torque to be added to the magnet torque.

In MTPA control, optimal d-and q-axes stator currents are determined in order to have a maximum torque. Applying negative d-axis current results in an advance angle γ as shown in Fig.3.5.1.1.

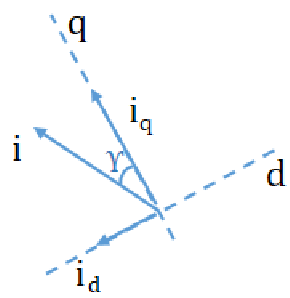

Figure 3.5.1.1: Advance Angle Resulted from a Negative d-axis Current

The torque equation can be rewritten in terms of γ as:

$$T = \frac{3P}{2}\Big[\lambda_{pm}|i|\cos(\gamma) - \frac{|i|^2}{2}\sin(2\gamma)(L_d - L_q)\Big] \quad (3.5.1.2)$$

Fig.3.5.1.2 depicts the behavior of output torque with respect to the advance angle γ.

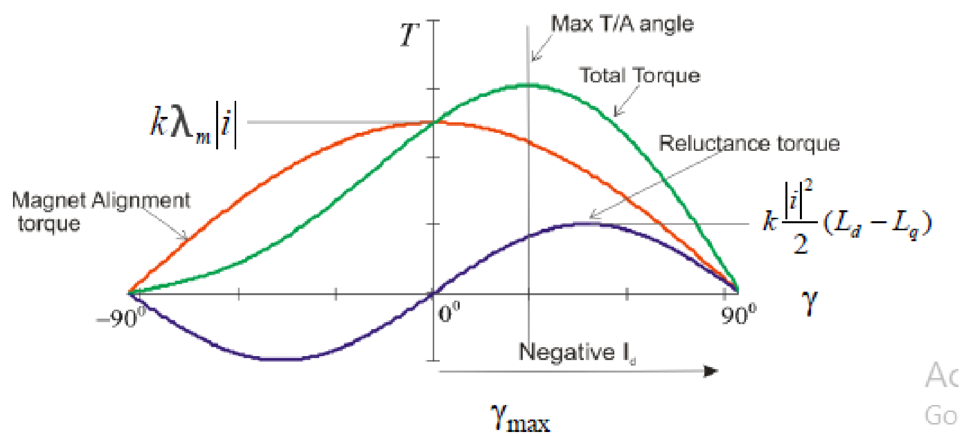

Figure 3.5.1.2: Torque Output as a Function of Advance Angle [4]

The advance angle γ_{maxT} at which the torque per current ampere becomes maximal can be calculated from;

$$\gamma_{maxT} = \sin^{-1}\Big(\frac{-\lambda_{pm} \pm \sqrt{\lambda_{pm}^2 + 8|i|^2\Delta L^2}}{-4\Delta L|i|}\Big) \quad (3.5.1.3)$$

The maximum torque is obtained by substituting the optimal angle into the torque equation. The relation between d-axis and q-axis currents for the maximum torque-per-ampere (MPTA) condition is given by,

$$id = \frac{\lambda_{pm}}{2(L_q - L_d)} - \sqrt{\frac{\lambda_{pm}^2}{4(L_q - L_d)^2} + iq^2} \tag{3.5.1.4}$$

Using the above relationships, a look-up table can be created with torque as the input and d and q axis currents as the outputs such that the lookup table outputs provide current references for the current controller.

3.5.2 Speed Controller Tuning

Bandwidth of the speed control loop is chosen to be sufficiently less than the bandwidth of the current control loop. In this way, the current controller will not have any influence on the speed controller performance. Furthermore, it will guarantee improved stability for the speed controller [3]. The sampling rate of the speed loop is selected different from that of the current loop to ensure independent execution of the two loops.

A block diagram of the speed control loop in S-domain is shown in Fig.3.5.2.1. Same as the current controller, PI controller is used for the design.

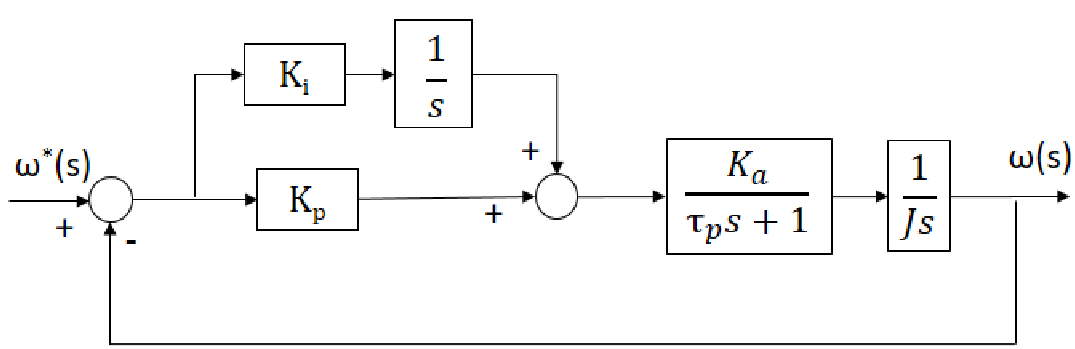

Figure 3.5.2.1: Speed Loop Model in S-domain

In this model, τ_p is the time constant of the speed loop and K_a is the torque constant of the motor.

In order to model the system in Z-domain, the effect of ZOH should be considered as explained in section 3.1.2. Using Eq.3.1.2.10, Z-domain controller transfer function C(z), with ZOH can be derived in to;

$$C(z) = \frac{K_p(z - \delta_c)}{(z - 1)} \tag{3.5.2.1}$$

where δ_c is the controller zero given by;

$$\delta_c = 1 - \frac{K_i T}{K_p} \tag{3.5.2.2}$$

T is the sampling time of the speed loop.

In the same way, plant transfer function P(z), in which speed loop and the mechanical model of the system are combined can be obtained as;

$$P(z) = \frac{mK_a}{J} \frac{(z - \frac{(T\delta_p - T + m)}{m})}{(z - 1)(z - \delta_p)} \tag{3.5.2.3}$$

where δ_p is the plant pole $e^{\frac{-T}{\tau_p}}$ and $m = T - \tau_p(1 - \delta_p)$

The OLTF of the speed loop can then be obtained as;

$$\begin{aligned} OLTF(z) =& C(s) * P(s) \\ =& \frac{mK_aK_p}{J} \frac{(z - \delta_c)(z - \frac{(T\delta_p - T + m)}{m})}{(z - 1)^2(z - \delta_p)} \end{aligned} \tag{3.5.2.4}$$

Gain k is defined as;

$$k = \frac{mK_aK_p}{J} \tag{3.5.2.5}$$

Controller zero δ_c is placed at a desired location preferably close to 1 (for example at 0.99). Root locus plot for the OLTF when current loop bandwidth is set to 300Hz is shown in Fig.3.5.2.2.

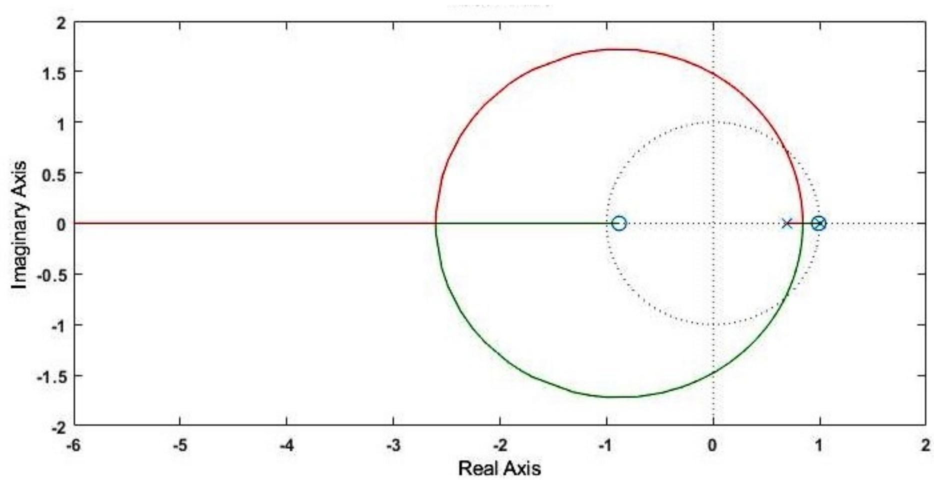

Figure 3.5.2.2: Root Locus Plot for Speed Loop in Z-domain

Fig.3.5.2.3 shows a zoomed view of the root locus plot shown in Fig.3.5.2.2.

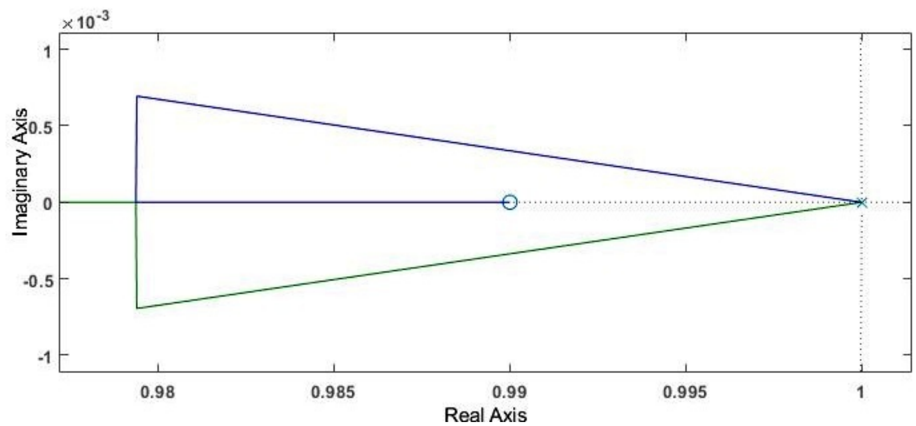

Figure 3.5.2.3: A Zoomed View of Root Locus Plot for Speed Loop in Z-domain

Gain k can be selected using root locus plot, so that the complex poles are avoided and the speed loop have the desired bandwidth.

Controller gains can then be calculated from;

$$K_p = \frac{Jk}{mK_a} \tag{3.5.2.6}$$

$$K_i = \frac{K_p}{T}(1 - \delta_c) \tag{3.5.2.7}$$

Chapter 4

Hardware and Software Setup

4.1 Hardware Setup

4.1.1 Motor Parameters

The motor to be controlled is an IPMSM machine (Model No.GK6025-8AF31-FE) whose parameters are given in Table 4.1.1.1. The motor is equipped with an incremental encoder.

Table 4.1.1.1: Motor Parameters

Parameter	Value
Rated Current	2.8 A
Torque Constant	0.45 Nm/A
Encoder maximum pulse count	10000
Polar Pairs	4
Back Emf Constant	0.0593 Vs/rad
Stator Resistance	2.44 Ω
D-axis Inductance	5.6 mH
Q-axis Inductance	7.52 mH
Rotor Inertia	1.073e-04 kg/m^2

APB22 type SPMSM machine is connected to the IPMSM machine as a load. Rotor inertia is calculated by summing up inertia of the two motors and inertia of the shaft connecting mechanical parts.

Offset angle and the back emf constant of the machine were estimated by experimental procedures which are described in the next subsections.

4.1.1.1 Offset angle calculation

In order to align the d-axis with the permanent magnet flux vector, the offset angle of the motor is required. The offset angle can be identified as the angle between the peak of phase A back-emf voltage and the zero pulse of the encoder.

The IPMSM is driven by an SPMSM to obtain the back emf voltages of the motor. Line to line voltages are measured as phase voltages of the motor are not directly accessible in the experimental setup.

In Fig.4.1.1.1, phase voltage with two L-L voltages are plotted for a typical system.

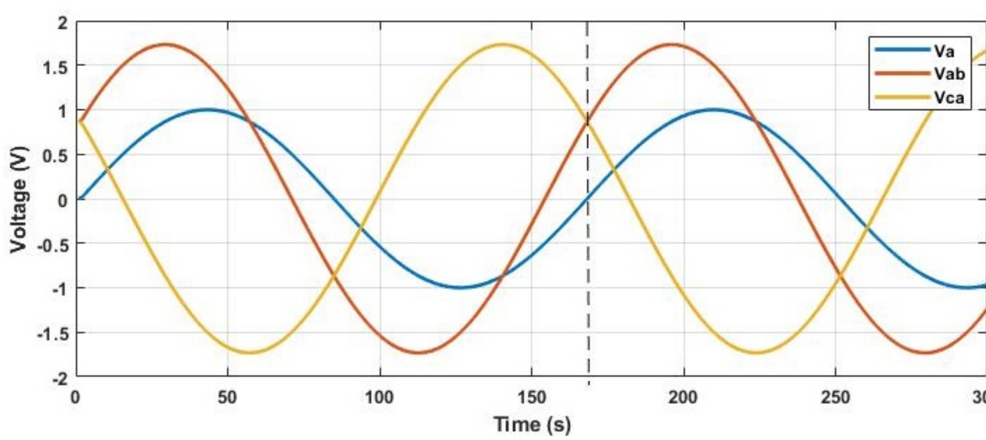

Figure 4.1.1.1: Phase and Line-Line Voltage Waveforms

From Fig.4.1.1.1, it is observed that the zero crossing of the phase A voltage occurs at the intersection point of the line to line voltages, Vab and Vca. Therefore, the position of the peak phase voltage can be obtained by shifting the intersection point of the two L-L voltages by 90^0.

The line to line voltages obtained experimentally are not pure sinusoidal waveforms. So, the data is imported to Matlab and Fast Fourier Transform (FFT) is performed in order to plot the fundamental voltage waveforms. The electrical angle obtained from encoder readings is used since zero-pulse occurs once per revolution. In Fig.4.1.1.2, the electrical angle is plotted along with the L-L voltages.

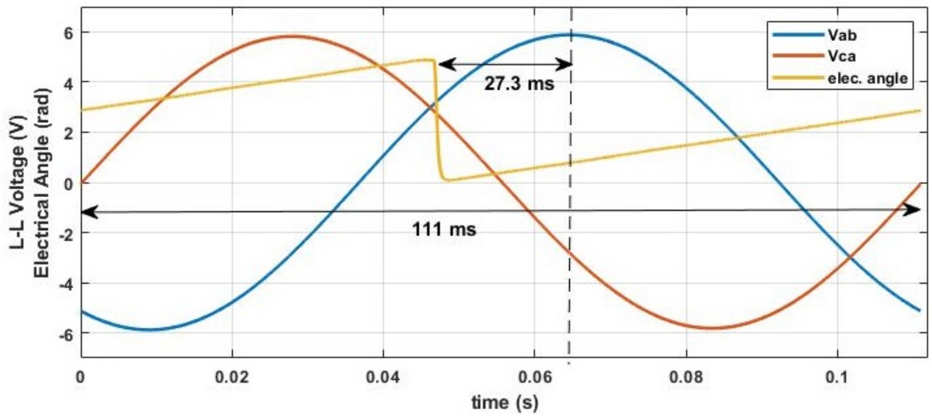

Figure 4.1.1.2: L-L Voltages and Electrical Angle to Calculate Offset Angle

Offset angle can be calculated as;

$$Offset\ angle = \frac{2\pi}{111} \times 27.3 = 1.5453\ rad = 88.54^0$$

Zero pulse leads peak back emf by 1.5453 rad. Therefore offset angle needs to be subtracted from the electrical angle of the rotor.

4.1.1.2 Back emf constant calculation

Back-EMF constant also known as flux linkage constant of the machine is determined by measuring peak phase voltages of motor in no-load condition at constant speed. Peak phase voltage can be calculated from the peak line-line voltage. Therefore, back-emf constant can be determined using the same experiment that is used to obtain the offset angle.

Dq-axis voltages are given by,

$$v_d = R_s i_d + \frac{d(L_d i_d)}{dt} - \omega_e (L_q i_q) \tag{4.1.1.1}$$

$$v_q = R_s i_q + \frac{d(L_q i_q)}{dt} + \omega_e(\lambda_{pm} + L_d i_d) \qquad (4.1.1.2)$$

Since zero current is flowing through windings, voltage appears on terminals is the q-axis voltage which results from the term $\omega_e \lambda_{pm}$. Therefore, the back-emf constant can be calculated as;

$$\lambda_{pm} = \frac{V_{pk}}{\omega_e} \times \frac{5.812 \times 0.111}{\sqrt{3} \times 2\pi} = 0.0593\, Vs/rad$$

4.1.2 High Voltage Motor Control and PFC Kit

A typical motor drive system running from AC power is illustrated in Fig.4.1.2.1.

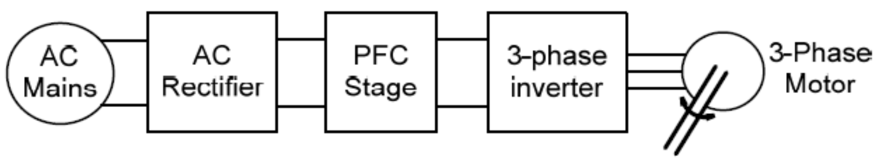

Figure 4.1.2.1: Block Diagram of a Motor Drive System with PFC Stage

High Voltage motor control and Power Factor Correction (HVCntrl+PFC) kit, developed by Texas Instruments, includes all the power and control blocks of a motor drive system. It is embedded with following main features;

- 3-Phase Inverter stage (350V DC max input voltage, 1KW/1.5KW max load)
- Power Factor Correction stage (750W max power rating, 400V DC max output voltage)
- AC Rectifier stage (85-132V AC/170-250V AC input, 750W max power rating)
- Auxiliary Power Supply Module (400V to 15V and 5V module)
- Isolated CAN interface
- Over-current protection for PFC stage

AC rectifier rectifies the AC voltage from the main. It can be used to either generate the DC bus voltage for the inverter directly or provide input for the Power

Factor Correction (PFC) stage present on the board. PFC stage enables wave shaping of the input AC current enabling efficient operation [12]. For the experimental work of this thesis, the inverter DC bus is directly fed by the rectifier without using the PFC stage. Auxiliary power supply module is used as a back-up power supply to provide DC bus voltage to the inverter. Gate signals to the inverter switches are generated by the microcontroller PWM signals.

The Printed Circuit Board (PCB) design and the schematic diagram of the motor control kit is shown in Fig.4.1.2.2 and Fig.4.1.2.3 respectively.

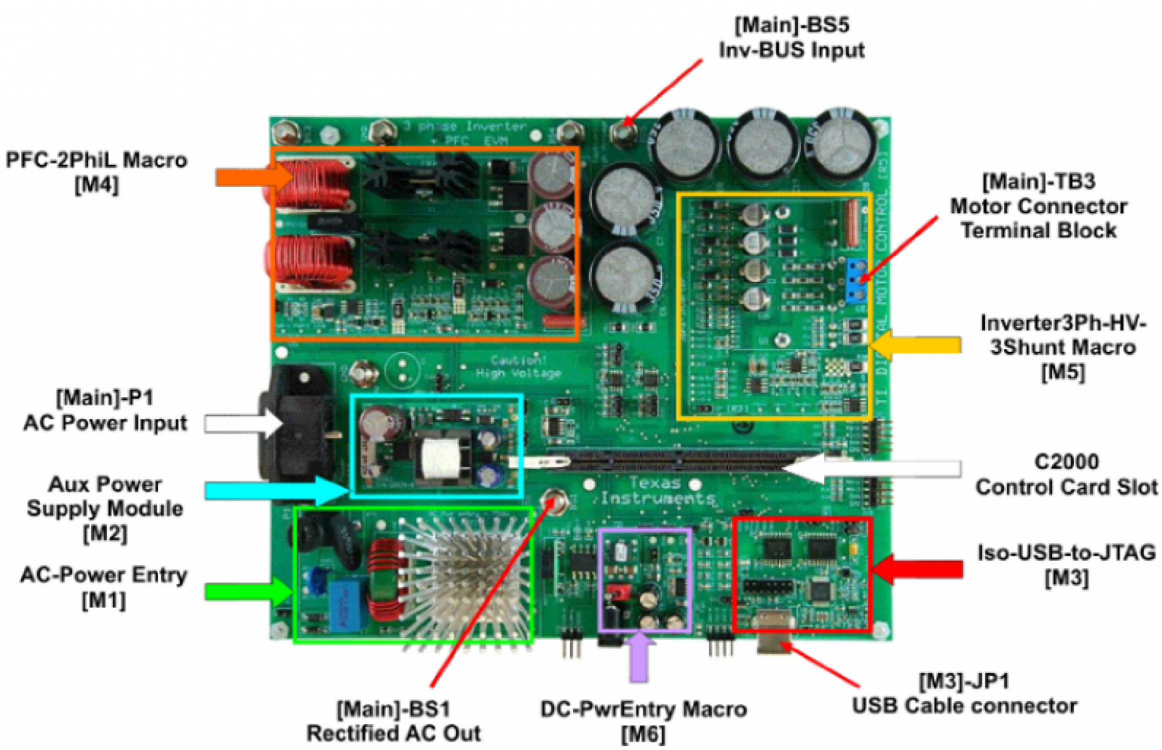

Figure 4.1.2.2: High Voltage Motor Control and PFC Kit [12]

As shown in the schematic diagram in Fig.4.1.2.3, there are several feedback signals which are measured from the ADC outputs of the micro-controller. Only inverter DC bus voltage and two phase current feedback signals are required to be measured for this experimental work.

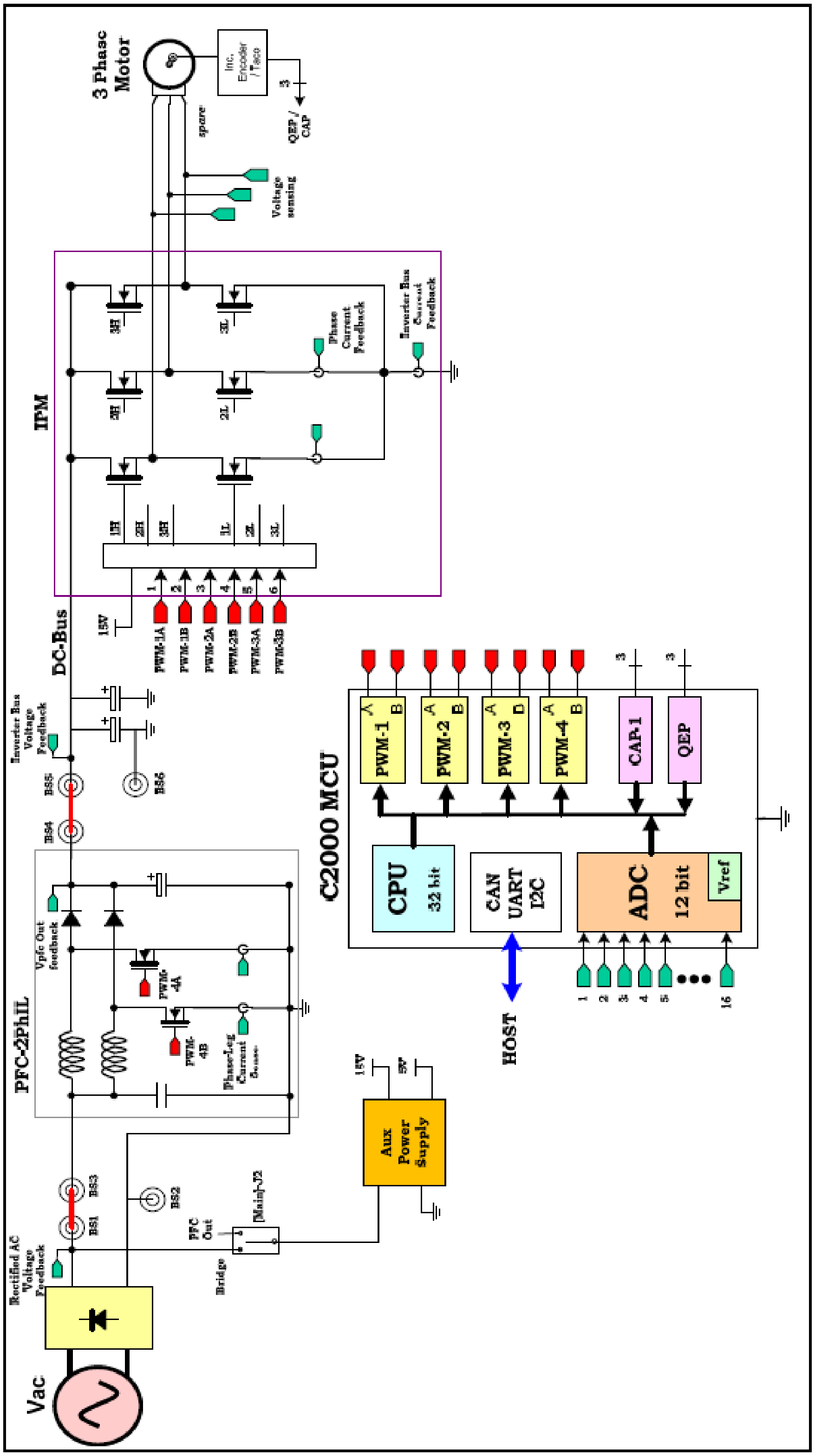

Figure 4.1.2.3: HV Motor Control and PFC Board Schematic Diagram with C2000 MCU [12]

4.1.3 Sensor Signal Conditioning

The ADC of the micro-controller handles signals from 0 to 3 V. Thus, a signal conditioning circuitry is embedded in the PCB design of the control kit in order to adapt the voltage and current signals into the required range.

4.1.3.1 Phase currents signal conditioning

The circuitry for adapting phase current signals is shown in Fig.4.1.3.1.

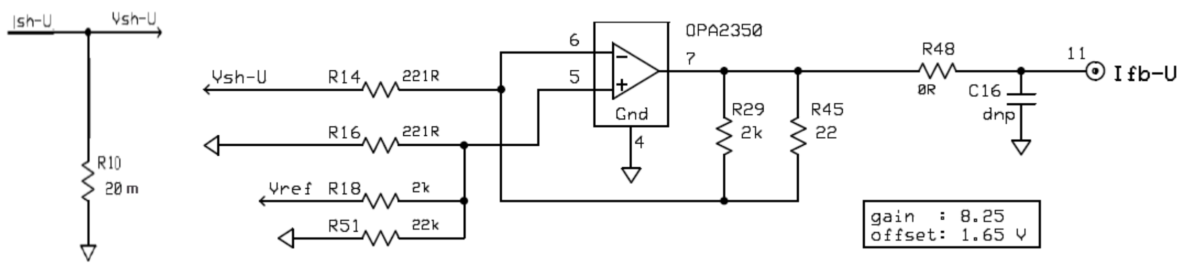

Figure 4.1.3.1: Signal Conditioning Circuit for Phase Currents

As depicted in Fig.4.1.3.1, sensed current signal is transformed into a voltage quantity which is measured through a 20mΩ resister. This improves sensitivity issues. The scaling circuit introduces a gain of 8.25 and an offset of 1.65 to the signal. Therefore, scaled phase current I_{fb} can be expressed as,

$$I_{fb} = 0.02 * 8.25 * I_{sh} + 1.65 \tag{4.1.3.1}$$

4.1.3.2 DC bus voltage signal conditioning

A simple voltage divider circuit has used to scale DC-bus voltage as shown in Fig.4.1.3.2.

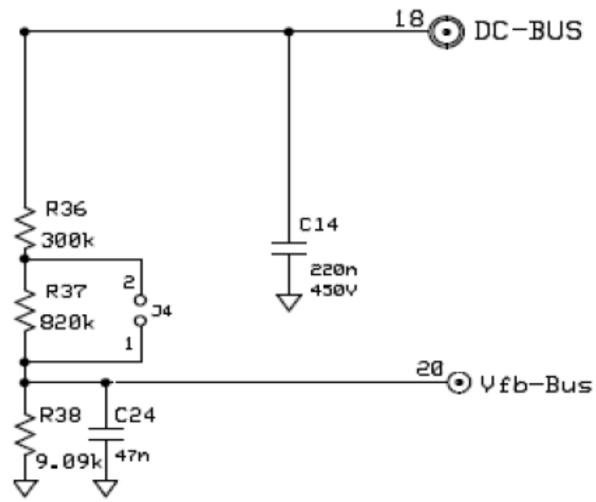

Figure 4.1.3.2: Signal Conditioning Circuit for DC-bus

Jumper J4 shown in Fig.4.1.3.2, facilitates the option to change the scaling factor. In our case, the jumper is kept open. Thus, the scaled DC bus voltage $V_{\text{fb-bus}}$ can be obtained as;

$$V_{fb-bus} = \frac{9.09k}{9.09k + 820k + 300k} V_{dc} \tag{4.1.3.2}$$

4.2 Software Setup

The control algorithm is developed using PSIM software. SimCoder is an add-on option of the PSIM software which can generate C code from PSIM schematics. The C code generated by SimCoder can run directly on target hardware platforms. Floating type TMS320F28335 microcontroller is used as the control card for the experimental work. Hardware target library is set in the SimCoder tab in the Simulation Control block as shown in Fig.4.2.1.

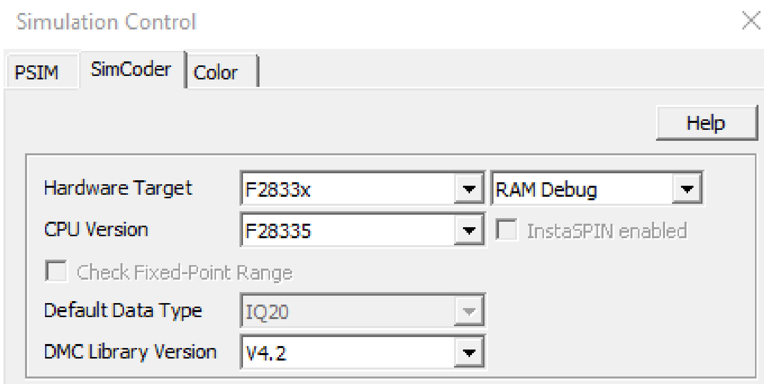

Figure 4.2.1: Hardware Target Configuration to use SimCoder Tool

It is important to mention that, code cannot be generated when control is in continuous domain. Therefore, the system should be developed in discrete-domain. Only the elements in the standard library that can be used for code generation are used and all the hardware elements are used from SimCoder F2833x hardware target section.

4.2.1 Hardware and Peripheral Configuration

F28335 provides 88 General Purpose Input Output (GPIO) ports (GPIO0 to GPIO87), and each port can be configured for different functions. The Hardware Configuration block is used for this purpose. Only one function can be selected for each GPIO port. For example, port GPIO0 can be used as either a digital input, a digital output or as a PWM.

The configuration of GPIO ports for the experiment are depicted in Fig.4.2.1.1. GPIO0-GPIO5 are assigned as PWM ports. A Serial Peripheral Interface (SPI) device is used to observe the signals through an oscilloscope. So, GPIO16-GPIO19 are allocated for the SPI ports. GPIO20, GPIO21 and GPIO23 are assigned for encoder signals (A,B and Z pulse). GPIO22 is used as a digital output to measure the code execution time if required.

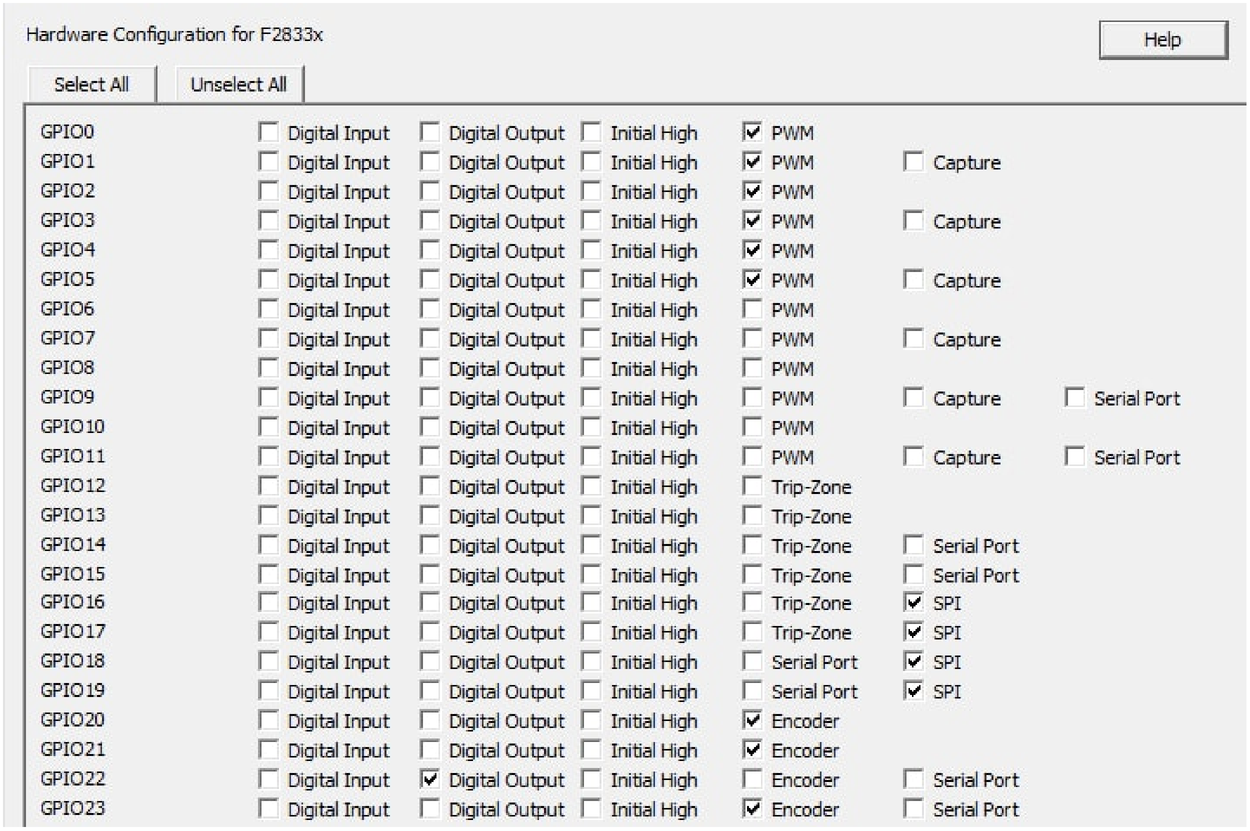

Figure 4.2.1.1: Configuration of GPIO Ports

4.2.1.1 Configuration of 3-phase PWM generator

F28335 provides 6 sets of PWM outputs. Each set has two outputs that are complementary to each other. For example, PWM 1 has a positive output PWM 1A and a negative output PWM 1B. Either PWM 123 or PWM 456 can be selected. In our case, PWM 123 is selected. PWM frequency scaling factor is set to 1 in order to have sampling frequency same as the PWM switching frequency.

Three-phase PWM block configuration is shown in Fig.4.2.1.2.

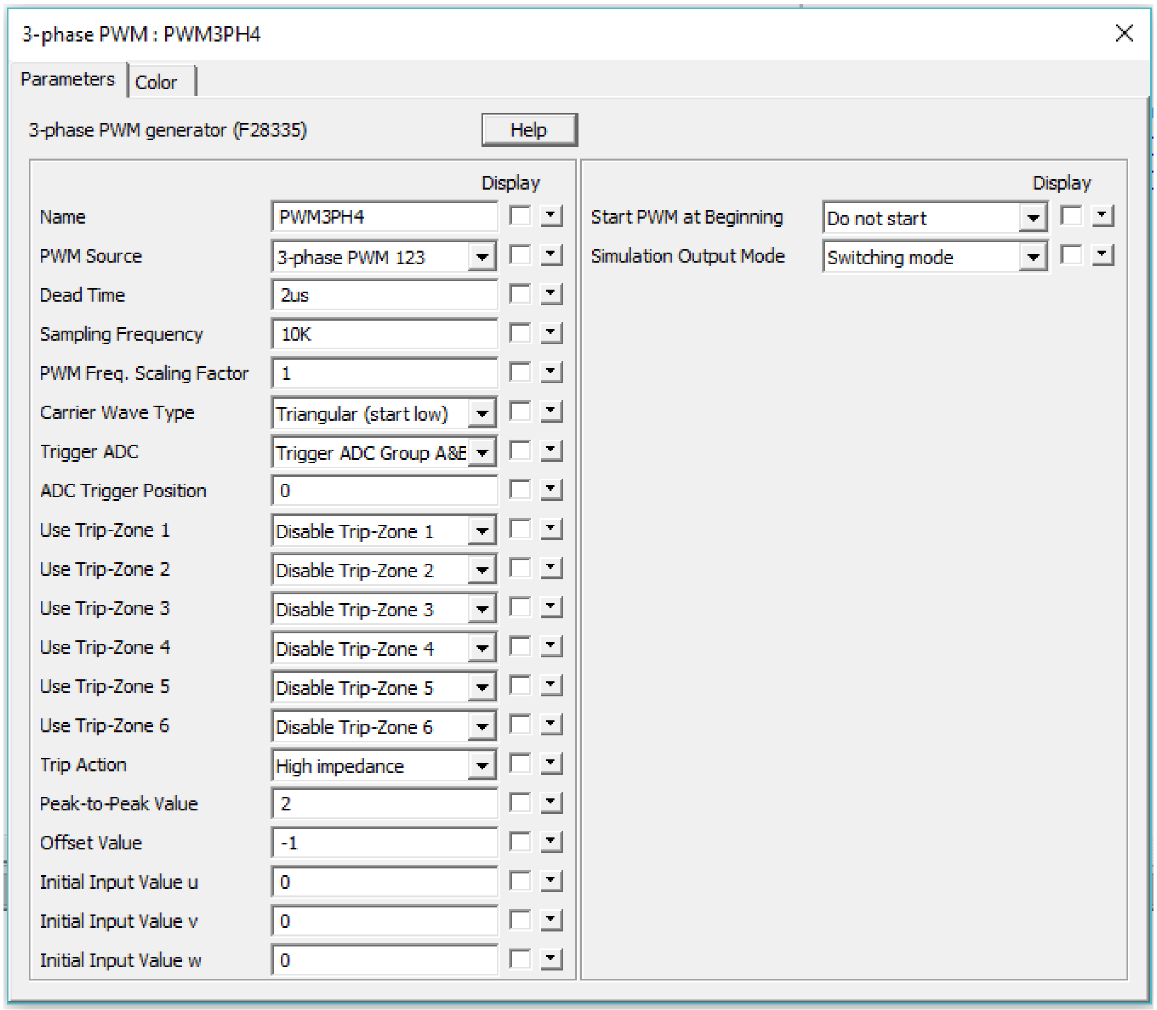

Figure 4.2.1.2: 3-phase PWM Generator Configuration

There are two types of carrier waveforms, triangular wave and sawtooth wave, each of them having two operation modes. The start mode depends on the way switch currents are measured. In a 3-phase inverter, if top switch currents are measured, "start-high" mode should be selected [13]. In the schematic shown in Fig.4.1.2.3, low switch currents are measured. Therefore, for the carrier wave type, "triangular (start-low mode)" is selected.

In start-low mode, the positive PWM output PWMA is low at the beginning of the switching cycle. The input and output waveforms of a PWM generator with the triangular carrier wave in start-low mode are shown in Fig.4.2.1.3.

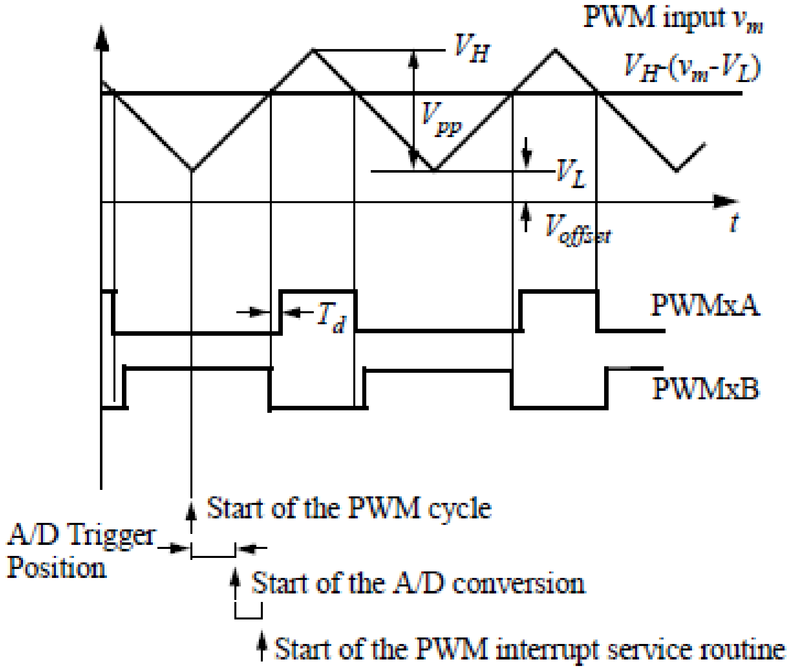

Figure 4.2.1.3: Operation of Start-Low Mode [13]

The Fig.4.2.1.3 also shows how the dead time is defined, and the time sequence when the PWM generator triggers the ADC. If triggering ADC is selected, A/D conversion will start after a certain delay defined by the A/D trigger position. Since ADC trigger position is set to zero in the block configuration, A/D conversion will start with the PWM cycle without any delay.

The PWM interrupt service routine starts after the A/D conversion is completed. If the PWM generator does not trigger the A/D converter, the PWM interrupt service routine will start at the beginning of the PWM cycle.

4.2.1.2 Start-Stop PWM Blocks Configuration

Initially, PWM is deactivated until starting command is applied through Code Composer Studio (CSS) window. "Start-PWM" and "Stop-PWM" blocks are used to facilitate PWM turning ON and OFF functionality. In configuration tabs, respective PWM source should be selected. To start PWM, a high logic signal should be applied to the input of the "Start-PWM" element. To stop PWM, a high logic signal should be applied to the input of the "Stop-PWM" element. This is implemented using the arrangement shown in Fig.4.2.1.4.

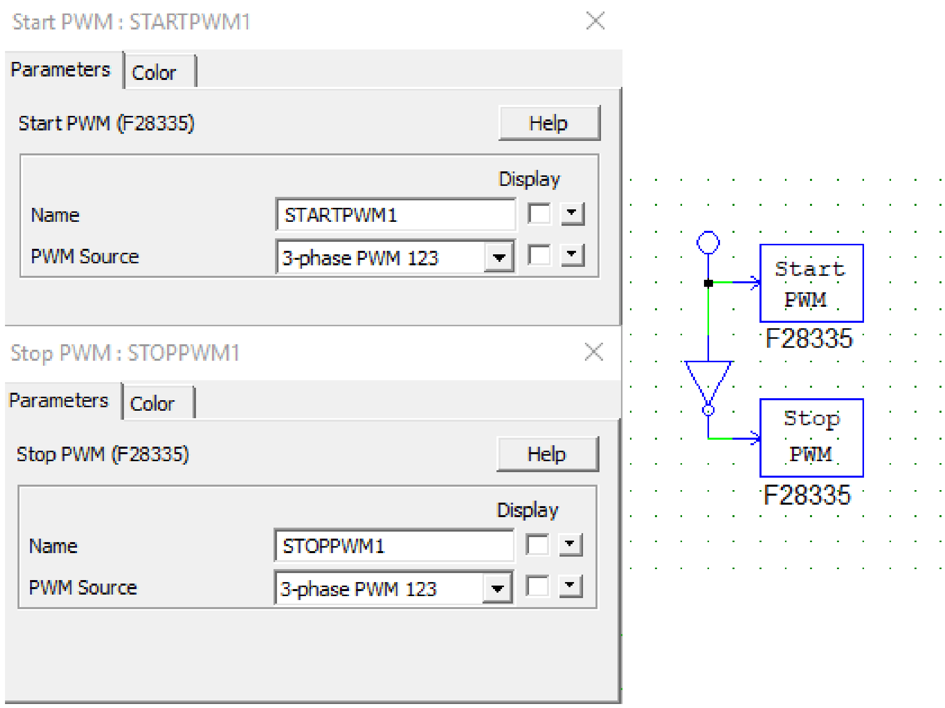

Figure 4.2.1.4: PWM Generation Control using Start-Stop Command

4.2.1.3 Configuration of Analog to Digital Converter

F28335 provides a 12-bit, 16-channel ADC which is divided into two groups, Group A and Group B. The input range of the ADC on the Digital Signal Processor (DSP) is from 0V to +3V. ADC module in the DSP can be operated in two sampling modes; Sequential sampling and Simultaneous sampling. In sequential sampling mode, 16 ADC input channels are sampled sequentially (i.e from A0 to A7 and then from B0 to B7). In simultaneous sampling mode, two ADC inputs, one from group A and one from group B (ex: A1 and B1) are sampled simultaneously. In PSIM Simcoder, sampling mode of ADC has set to sequential mode.

ADC resources allocation for phase currents and DC bus voltage signals are given Table 4.2.1.1.

Table 4.2.1.1: ADC Resources Allocation

Signal Name	ADC Channel No Mapping	Function
Ifb-U	ADC-B3, A1	Low side U-phase current sense
Ifb-V	ADC-B5, B1	Low side V-phase current sense
Ifb-W	ADC-A3, A5	Low side W-phase current sense
Ifb-bus	ADC-A7	DC Bus Voltage sense

The motor windings are wye connected without a neutral, so, Iu + Iv + Iw = 0. Therefore, if two phase current are measured the third one can be computed.

For efficient vector control implementation, current signals should be sampled at the same time. Since Simcoder ADC samples signals sequentially, it is not possible to have two current signals sampled at the same time. Therefore, the ADC channels are selected so that the current signals are sampled with minimum time lag.

ADC channel B3 and B1 are selected to read phase U and phase V currents respectively. For DC-link voltage, there is no such issue to be considered thus it is read from ADC channel A7 as shown in Fig.4.2.1.5.

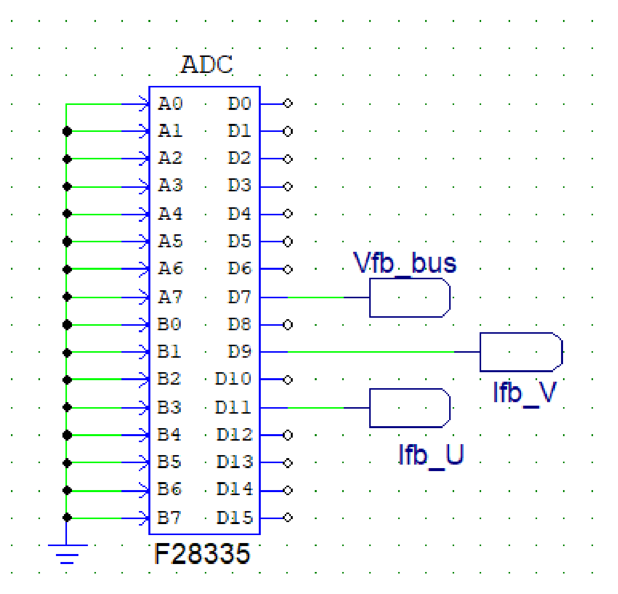

Figure 4.2.1.5: Measuring ADC Output Signals

There are three modes of operation of the ADC in Simcoder.

- Continuous: A/D conversion is performed continuously.
- Start/stop (8-channel): A/D conversion is only performed upon request, on only one of the 8-channel groups.
- Start/stop (16-channel): A/D conversion is only performed upon request, on all 16 channels.

The two "Start/Stop" modes can only be used when ADC is triggered by a PWM generator. In our case, ADC is triggered by the PWM generator, thus "Start/Stop (16-channel)" mode is selected. DC channel mode is used, because the input signal range is from 0 to +3V. The channel gain is set to unity.

There was an issue with the sampling rate of the ADC when the "Start/Stop" mode is selected. Since we were unable to solve the issue, we informed the PSIM support team. Their guess was it might be due to overflowing of Interrupt Service Routine (ISR). However, after checking the code execution time, it was realized that there was no such issue. It was because of a bug in SimCoder and the PSIM support team instructed to solve it by myself till they release a patch.

4.2.1.4 Configuration of Encoder

F28335 has 2 Encoders. Encoder 1 can be at either Port GPIO 20-21, or Port GPIO50-51. Encoder 2 is at Port GPIO 24-25. Encoder 1 at Port GPIO20-21 and at Port GPIO50-51 uses the same inner function blocks. Therefore, they cannot be used at the same time.

An incremental encoder which is described in section 1.2.1 is used to measure the rotor position. Encoder block in Simcoder is modelled such that it can determine the rotor position when A, B and Z signals are given as inputs. Either index mark (Z) or Strobe input can be used to latch the encoder's initial position. The counting direction can be either Forward or Reverse. When it is set to Forward, the encoder counts up. Otherwise, the encoder counts down. Encoder resolution is the maximum pulse count of the encoder.

Encoder configuration is shown in Fig.4.2.1.6.

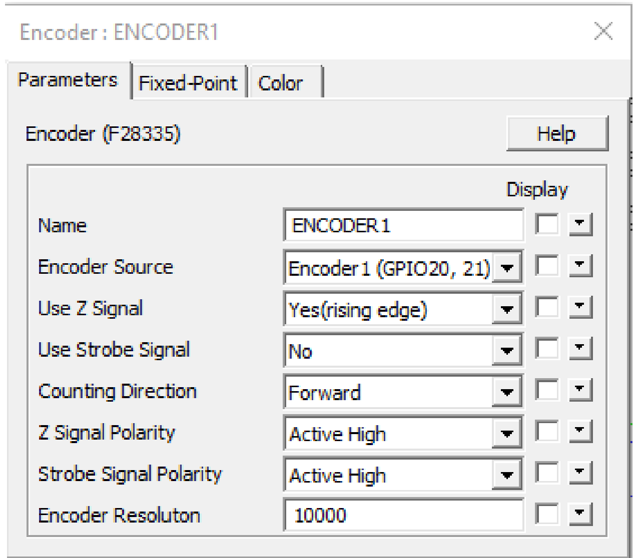

Figure 4.2.1.6: Configuration of Encoder

The Encoder Index Position block is used to latch the encoder's initial position. When the input of this block is 0, the encoder counter is set to 0. The encoder enables the latch when the input changes to 1 and latches the counter value when it meets the index event.

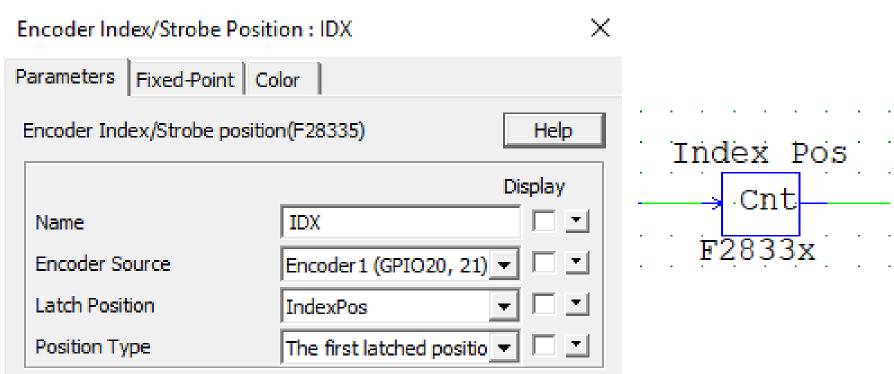

Figure 4.2.1.7: Configuration of Index Position Element

The objective of using the index position block in our model is to detect the first index pulse which is required for the motor alignment procedure discussed in chapter 5. In the configuration tab, Encoder source used in the model is selected. Latch position is set to "indexpos" as Index/Z signal is used in encoder configuration. The first latched position is selected to detect the first index pulse.

4.2.1.5 Serial Peripheral Interface Configuration

Serial Peripheral Interface blocks can be used to implement the function to communicate with external SPI devices (such as external A/D and D/A converters) easily and conveniently. There are two sets of GPIO ports, GPIO 16-19 and GPIO 54-57 which can be used as SPI ports. SPI device pin allocation for the GPIO 16-19 ports are given in Table 4.2.1.2 [14].

Table 4.2.1.2: PIN Allocation for SPI device

GPIO port	SPI pin
GPIO16	data output pin
GPIO17	data input pin
GPIO18	clock SPICLK
GPIO19	slave transmit-enable pin SPISTE

The SPI Configuration block defines chip selection pins and the buffer size for the SPI commands. F28335 supports up to 16 SPI devices, which requires four GPIO pins for chip select as defined by Chip Select Pin0 to Pin3. These GPIO ports and the SPI slave transmit-enable pin SPISTE are used to generate the chip select signal for SPI devices. If there is only one SPI device, the SPI slave transmit-enable pin SPISTE can be used as the chip select signal.

Each memory cell of the buffer saves the index of an SPI command. Normally, buffer size can be specified as 1 plus the number of SPI commands (i.e. Start Conversion Command, Receiving Data Command, Sending Data Command, and Sync. Command) in all SPI Input/Output elements.

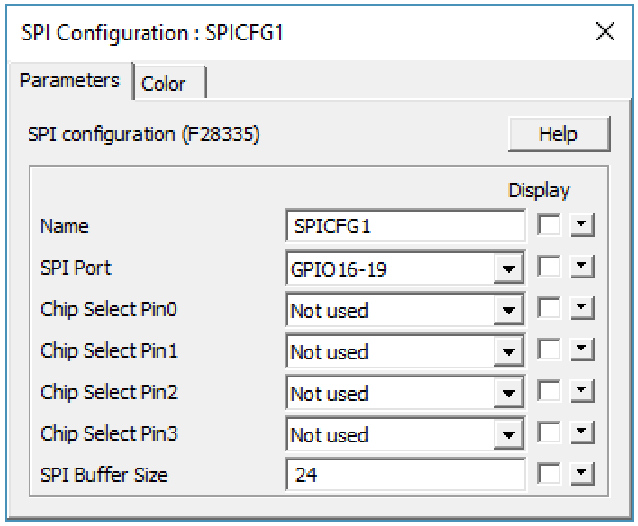

Figure 4.2.1.8: Settings for SPI Configuration Block

As shown in Fig.4.2.1.9, the chip select pins of the SPI device are connected to the chip select pins of the SPI Configuration block to ensure that settings made, are for the respective device. [15]

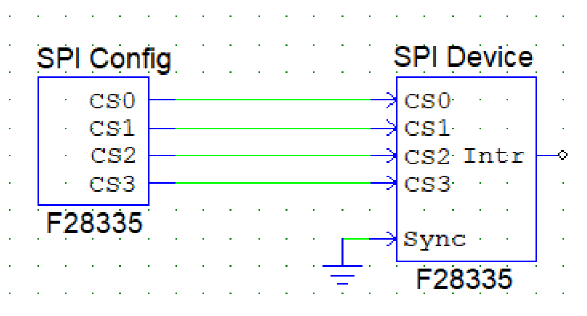

Figure 4.2.1.9: SPI device and Configuration Block Connection

AD7568 Digital to Analog Converter is used as the SPI device to observe some signals of interest. SPI Device block defines the information of the corresponding SPI hardware device.

The AD7568 is a serial input device and it contains eight 12-bit Digital to Analog Converter (DAC)s in one monolithic device. Data is loaded using $\overline{FSIN}$, CLKIN and SDIN. One address pin, A0, sets up a device address, and this feature may be used to simplify device loading in a multi-DAC environment. [16]

The communication speed of the device is 10MHz. Data is received on the rising edge of the clock without any delay. FSIN Level-Triggered Control Input (Active Low) is the frame synchronization signal for the input data. When FSIN goes low, it enables the input shift register, and data is transferred on the falling edges of CLKIN. If the address bit is valid, the 12-bit DAC data is transferred to the appropriate input latch on the sixteenth falling edge after FSIN goes low.

The device accepts a 16-bit word, SDIN. The first bit (DB15) is the MSB, with the remaining bits following. Next comes the device address bit, A0. If this does not correspond to the logic level on Pin A0, the data is ignored. Finally comes the three DAC select bits. These determine which DAC in the device is selected for loading.

Fig.4.2.1.10 shows the configurations for the SPI device.

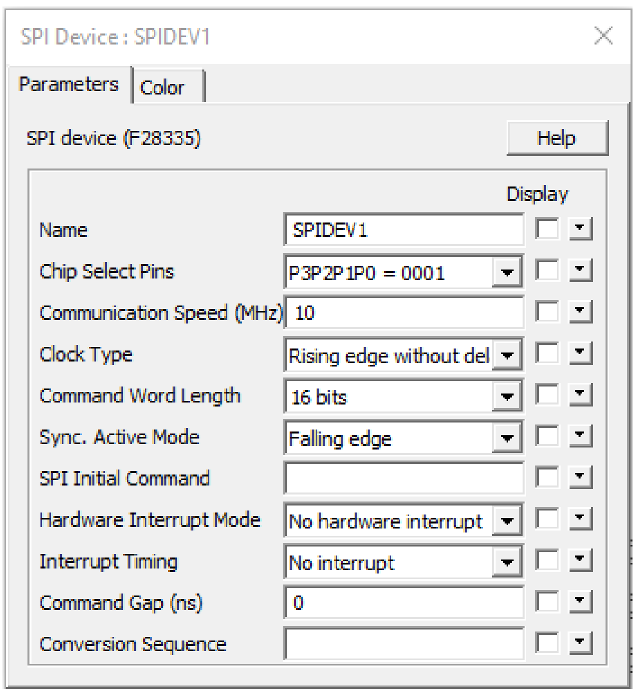

Figure 4.2.1.10: Configurations for SPI device

The DSP signal which needs to measure through an output channel of the DAC, is connected to an SPI output element. The properties of an output channel is defined in the SPI output block. One SPI output block corresponds to one output channel. AD7568 DAC has 8 output channels, thus eight DSP signals can be displayed simultaneously using eight SPI output devices. Settings for an SPI Out block are shown in Fig.4.2.1.11.

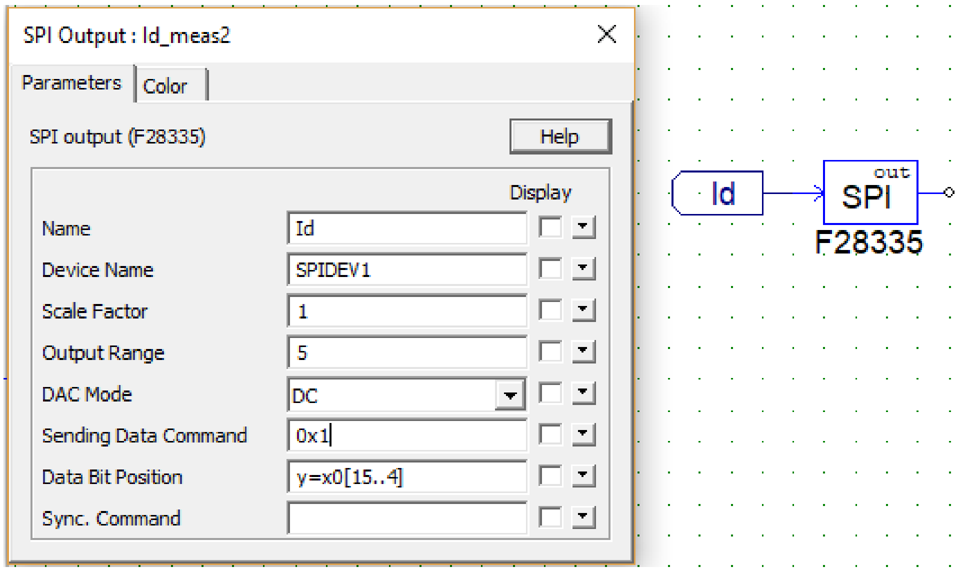

Figure 4.2.1.11: Settings for an SPI Out Element

Output Range of the device is 0-5V. If the signal to be displayed, has negative values or amplitude higher than 5V then its amplitude must be shifted and scaled to fit within the output range of the DAC. Sending Data Command specifies the output channel. For example, command 0x1 is to select Channel 1. Data Bit Position defines the position of the 12 bits data in the 16 bits sending data string.

4.2.2 Determining Real-time DSP Signals

As explained in section 3.1.3, voltage and current signals are adapted to the range of 0-3V using signal conditioning circuits. Therefore, the following relationships are used to determine actual DSP signals from the ADC outputs.

$$V_{dc} = \frac{9.09k + 820k + 300k}{9.09k} \times V_{fb-bus} \tag{4.2.2.1}$$

$$I_{sh} = \frac{(I_{fb} - 1.65) \times 50}{8.25} \tag{4.2.2.2}$$

Only two phase currents are measured at the ADC output, thus third current I_w is computed from;

$$I_u + I_v + I_w = 0 \tag{4.2.2.3}$$

Fig.4.2.2.1 shows the PSIM schematic for determining phase currents and DC voltage signals.

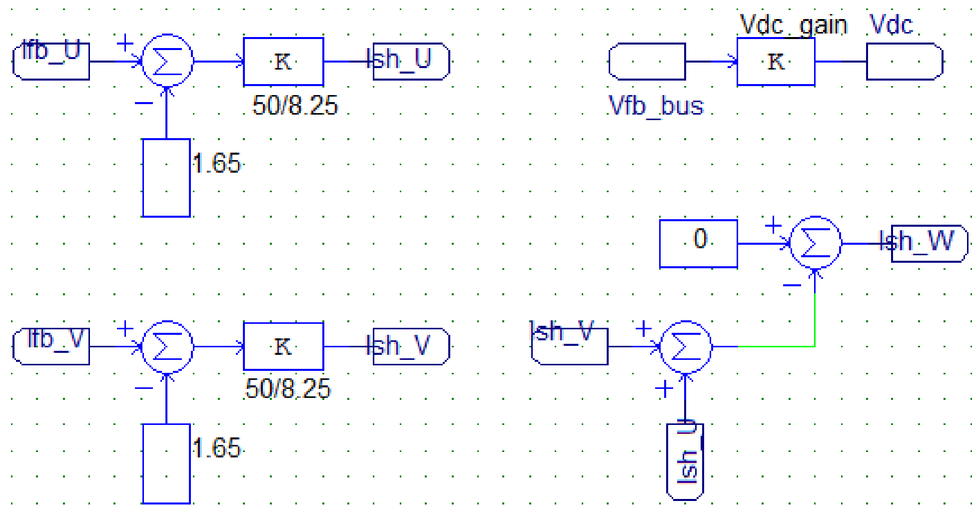

Figure 4.2.2.1: Scaling ADC Output Current and Voltage Signals

4.2.3 Rotor Mechanical and Electrical Angle Calculation

Encoder output counts from 0 to 10000 (encoder resolution) and resets to 0 when the Z pulse is received. Mechanical angle is obtained from the encoder output as shown in Fig.4.2.3.1.

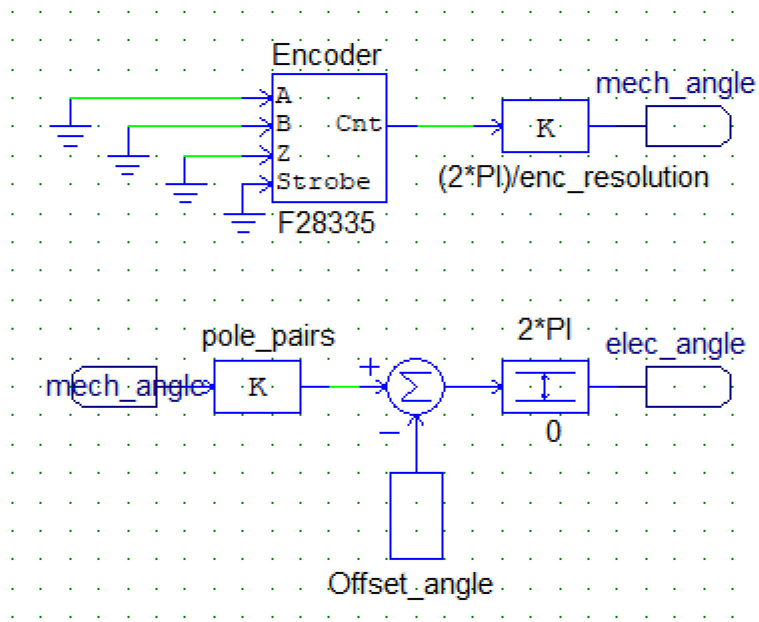

Figure 4.2.3.1: Obtaining Rotor Mechanical and Electrical Angle

Electrical angle can be obtained by multiplying mechanical angle with the number of pole pairs. The offset angle calculated in section 4.1.1.1 is subtracted from the electrical angle to have the correct d axis alignment. The limiter is used to make the electrical angle between 0 and 2π.

4.2.4 Axis Transformation

Three-phase current signals are transformed into dq synchronous reference frame currents using the Eq.4.2.4.1. Dq axis orientation of abc to dq0 transformation block which is available in PSIM library is different from ours. Therefore the schematic shown in Fig.4.2.4.1 is developed according to the selected dq axes orientation.

$$\begin{bmatrix} i_q \\ i_d \\ i_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \qquad (4.2.4.1)$$

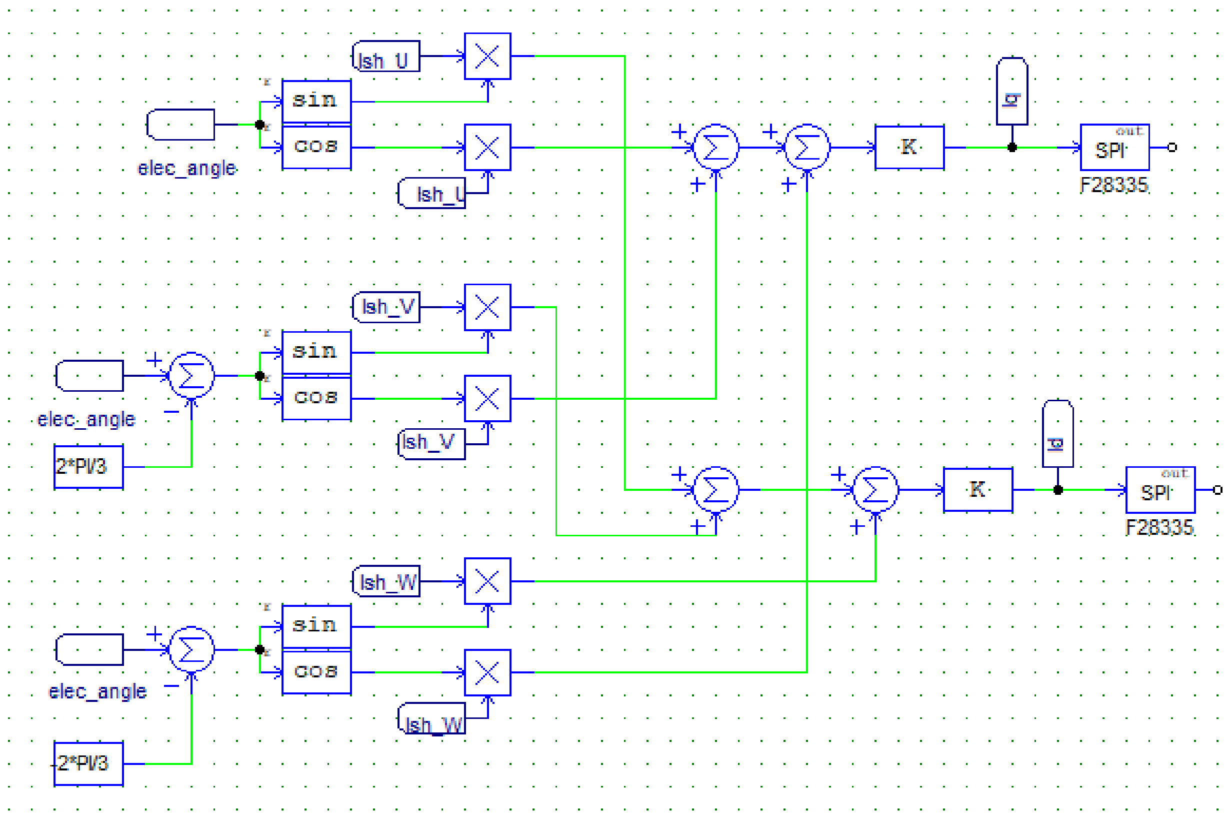

Figure 4.2.4.1: ABC to DQ transformation

Chapter 5

Development of Control Schematics

5.1 Rotor Alignment at Start-Up

Incremental encoders are usually preferred in speed control applications due to their low cost and simple structure. However, when an incremental encoder is used, the initial position of the rotor is unknown as it measures only the change in position. Position value of the incremental encoder at start-up is illustrated in Fig.5.1.1.

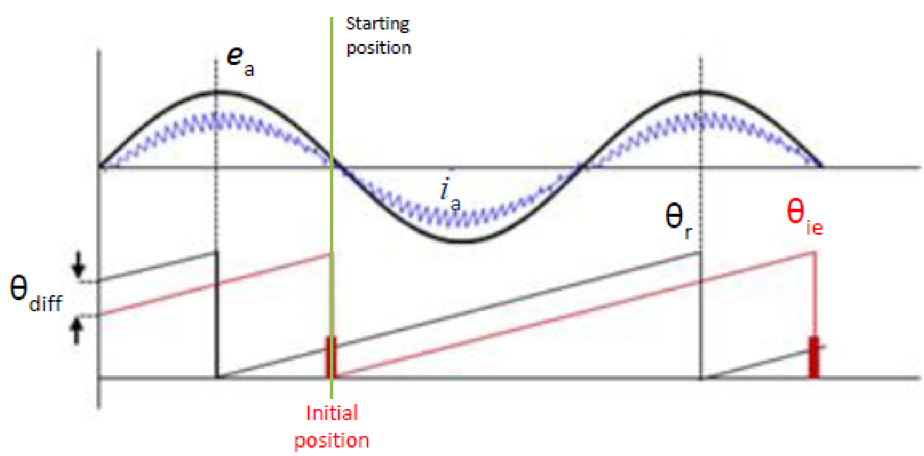

Figure 5.1.1: Incremental Encoder Position with Actual Rotor Position [17]

In Fig.5.1.1, θ_r is the real rotor position, θ_{ie} is the incremental encoder position.

θ_{diff} is the position error. The zero position of the encoder is aligned with the zero-crossing position of phase-A stator voltage. The position error will change according to the starting position. Therefore, when an incremental encoder is used, either the position error must be known and added to the incremental encoder output, or the rotor must be brought to zero position before starting the control algorithms [17].

In our case, the rotor is brought to zero position by applying phase voltages to stator windings sequentially for one full rotation of the rotor. A fixed voltage vector is applied for 0.5s along phase a,b and c sequentially. This allows the rotor to rotate in a quantized fashion, until the zero pulse is detected. This procedure is to be executed only at start-up, i.e. before closed-loop control is enabled.

Implementation is done using event handling concept in PSIM Simcoder. In the schematic shown in Fig.5.1.2, there are two states: S1 and S2 both in the form of sub-circuits. Sub-circuit S1 is modelled for rotor alignment at start-up. It includes a C Coding block in which alpha-beta voltages are generated as inputs to a space vector generation block to calculate space vector time durations (see listing A.1). S1 is defined to be the default state by the connection of the default event element to the input event port. Thus it is executed initially.

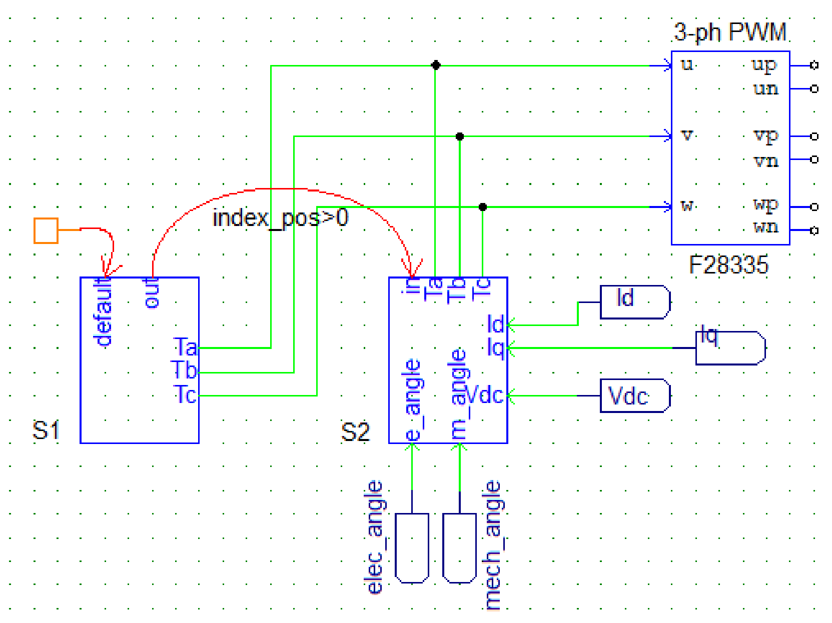

Figure 5.1.2: Using Sub-Circuits for Event Control

Sub-circuit S2 includes the schematic model for closed-loop control of the motor. The system will transit from state S1 to S2 when the condition 'index-pos>0' is met i.e when the index pulse is received. Index pulse is detected using index position block as shown in Fig.5.1.3. The output of the index position is 0 until the index is detected. Then it gives the encoder count at which the index is received. At that point of time, encoder resets and thereafter absolute position of the rotor is known.

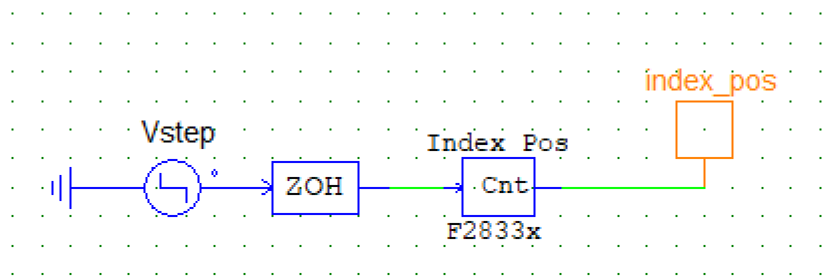

Figure 5.1.3: Index Pulse Detection using Index Position Element

It is important to mention that all the hardware elements i.e ADC, PWM block, Encoder, etc. should be placed in the main circuit. They cannot be placed inside a sub-circuit.

5.2 Current Control Loop

The control schematic is developed inside the subsystem S2. If the input of a subsystem has a sampling rate, and the rate can not be derived from the circuit inside the subsystem, a zero-order-hold block must be connected at the input to explicitly define its sampling rate [13]. However, to avoid ambiguity, a zero-order-hold block is placed at each input to define its sampling rate. The sampling frequency of the current control loop is the same as PWM switching frequency, 10kHz.

5.2.1 Cross-Coupling Decoupling Current Controller

The schematic for cross-coupling decoupling synchronous frame current regulator with back-emf compensation and anti-windup is shown in Fig.5.2.1.1.

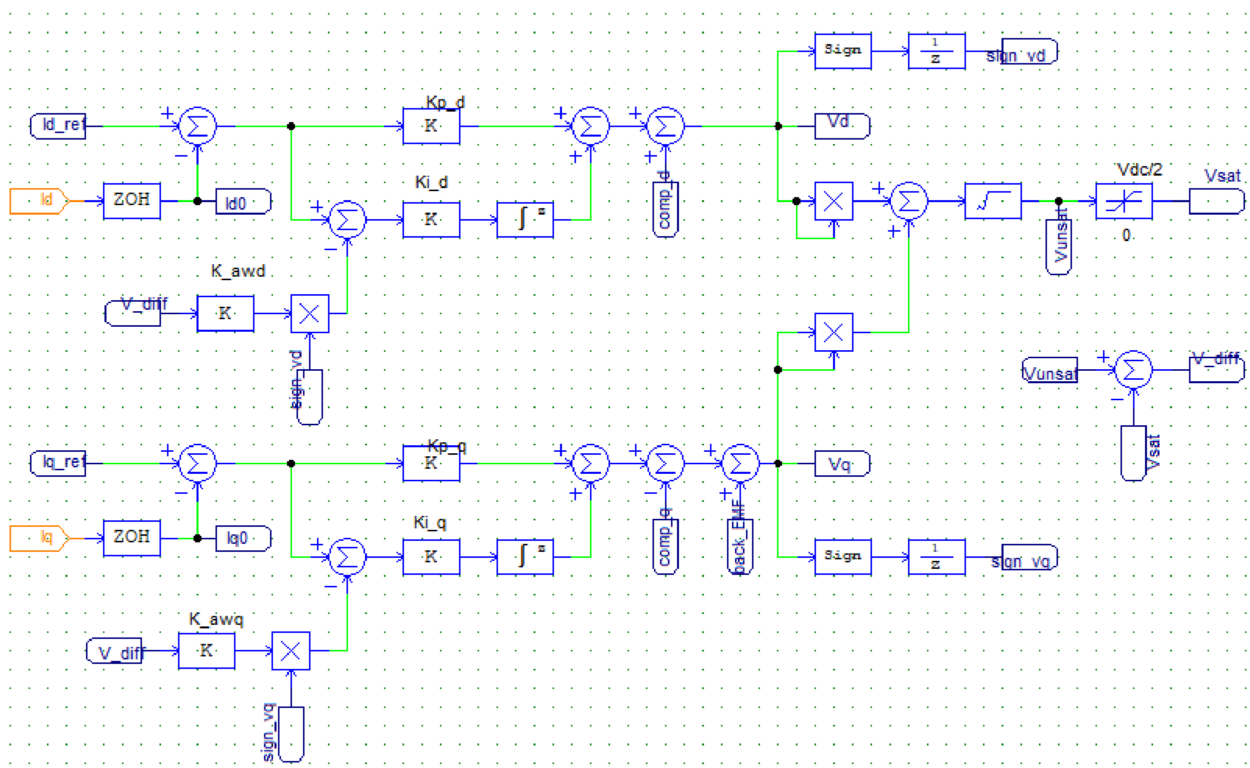

Figure 5.2.1.1: Schematic Diagram for Cross-Coupling Decoupling Current Controller

The schematic for obtaining dq axis decoupling and back-emf compensation components is shown in Fig.5.2.1.2. They are obtained according to the d-axis and q-axis voltages given by;

$$v_q = R_s i_q + \frac{d(L_q i_q)}{dt} + \omega_e \lambda_{pm} + \omega_e L_d i_d \tag{5.2.1.1}$$

$$v_d = R_s i_d + \frac{d(L_d i_d)}{dt} - \omega_e L_q i_q \tag{5.2.1.2}$$

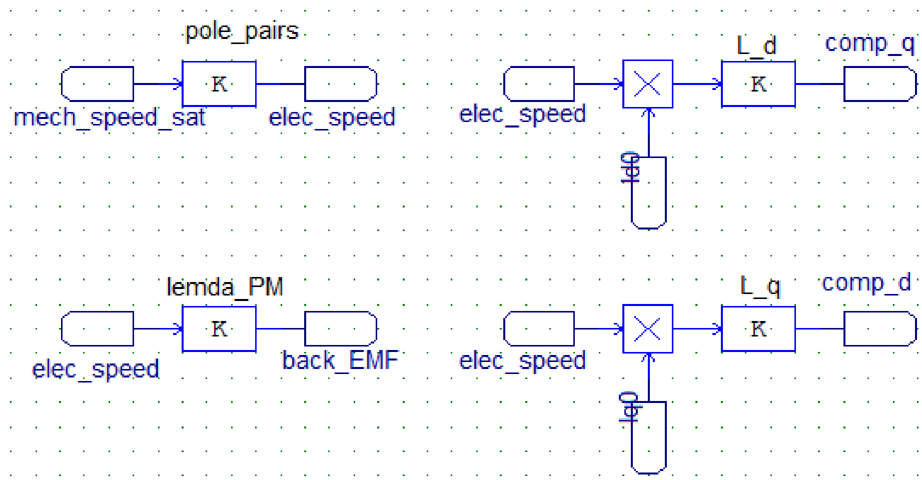

Figure 5.2.1.2: Back EMF and Decoupling Components

The back emf constant of the machine is obtained experimentally as discussed in section 4.1.1.2. Enhanced mechanical speed obtained from the vector tracking observer is used to calculate the electrical speed of the rotor.

Anti-wind up scheme is implemented based on the back-calculation algorithm by considering the voltage reference saturation limits. The difference between the regulator output voltage (V_{unsat}) and the voltage after the saturation limiter (V_{sat}) is fed back to the integrator input along with the signs of q and d axis voltages.

5.2.2 Complex Vector Current Controller

The schematic of the complex vector current regulator is shown in Fig.5.2.2.1. As discussed in section 3.1.1.2, in this design, dq axis decoupling is implemented inside the PI controller avoiding the requirement for using the system parameters L_d and L_q.

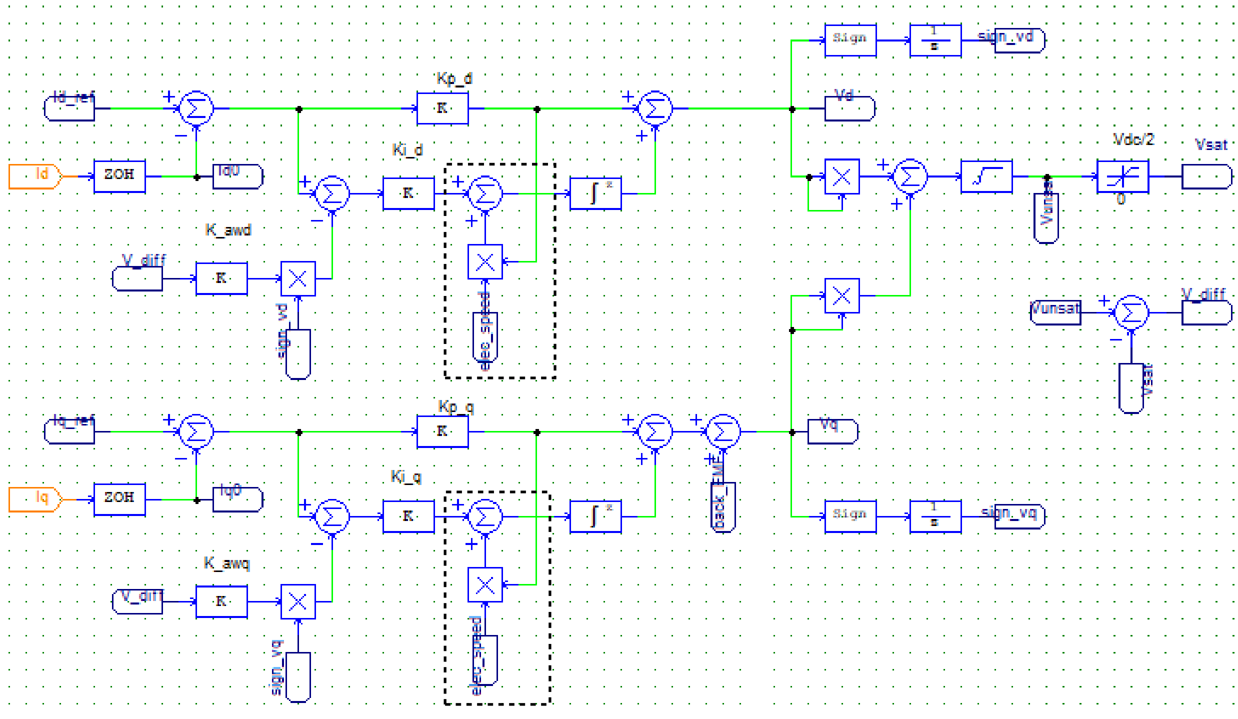

Figure 5.2.2.1: Schematic Diagram for Complex Vector Current Regulator

5.2.3 Drive Start-Up and Dead Time Compensation

Space vector PWM technique which is discussed in section 1.3.2. is used for this thesis work. Inputs to the space vector generation block in the PSIM library, are reference voltages in alpha-beta stationary reference frame. Therefore, dq axis voltages are converted into voltages in alpha-beta reference frame according to;

$$\begin{bmatrix} v_\alpha \\ v_\beta \\ v_0 \end{bmatrix} = \begin{bmatrix} \sin\theta & \cos\theta & 0 \\ -\cos\theta & \sin\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_d \\ v_q \\ v_0 \end{bmatrix} \tag{5.2.3.1}$$

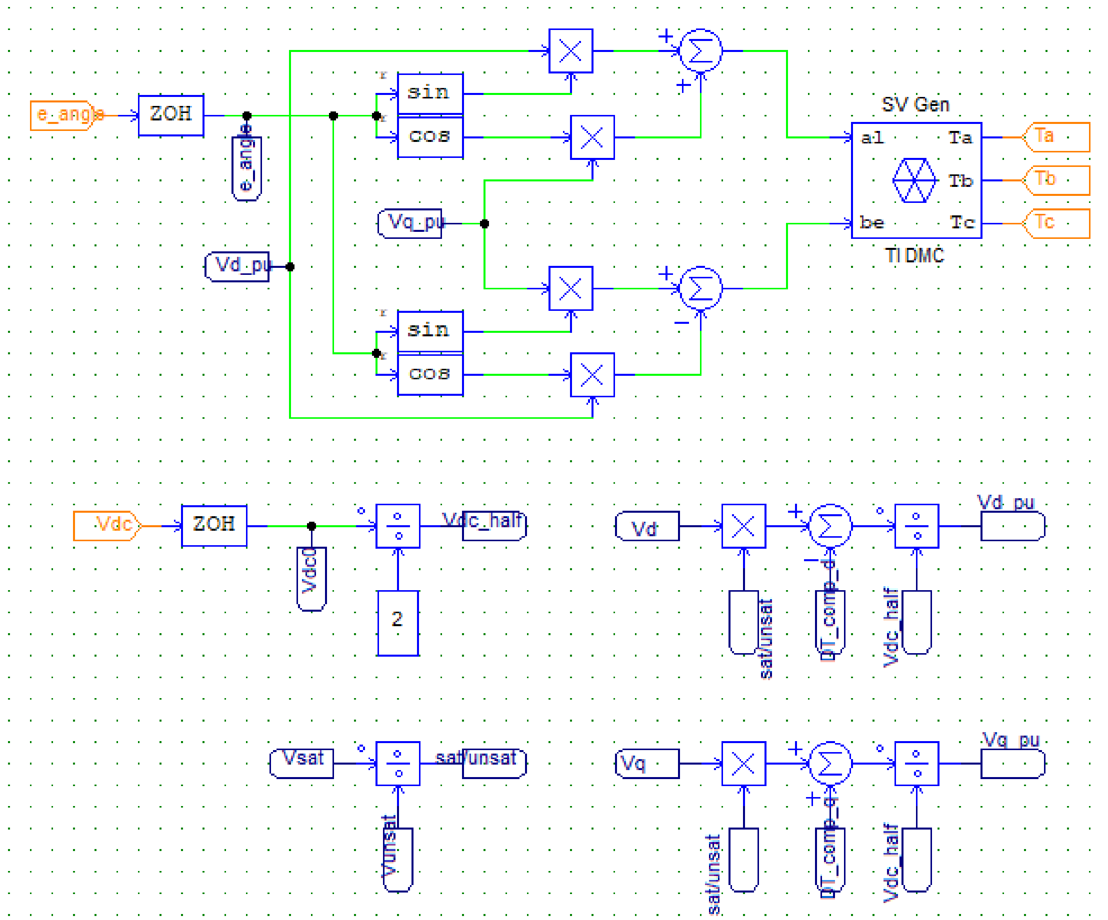

Figure 5.2.3.1: Alpha-beta Voltage References Calculation and Dead-time Compensation

When the voltage is saturated, there is a difference between saturated and unsaturated voltages. Thus q and d axis voltage references need to be scaled using the ratio between saturated and unsaturated voltages. The ratio will be unity when voltage references are below the saturation limit.

The amplitude of the carrier wave is set to unity when configuring the PWM block. Therefore, per unit values of the voltages are calculated based on DC link voltage.

In order to avoid the effect of dead time, dead time compensation voltages are added to the dq axis reference voltages. As mentioned in section 3.3, dead time compensation voltages given in Eqs.3.3.1-3.3.2 are manipulated according to the selected dq axis orientation as follows;

$$V_d^* = -\frac{4}{3} V_f \sin\left[\frac{\pi}{3} Trunc\Big(\frac{\omega_e t + \theta^* + \frac{\pi}{6}}{\frac{\pi}{3}}\Big) - \omega_e t\right] \tag{5.2.3.2}$$

$$V_q^* = \frac{4}{3} V_f \cos\left[\frac{\pi}{3} Trunc\Big(\frac{\omega_e t + \theta^* + \frac{\pi}{6}}{\frac{\pi}{3}}\Big) - \omega_e t\right] \tag{5.2.3.3}$$

$V_f = V_{dc}(T_{dt}/T_s)$ and $\theta^* = \tan^{-1}(i_q^*/i_d^*)$ and ω_e is electrical angular frequency of synchronous reference frame

The schematic for calculating dead time compensation voltages in dq reference frame is shown in Fig.5.2.3.2.

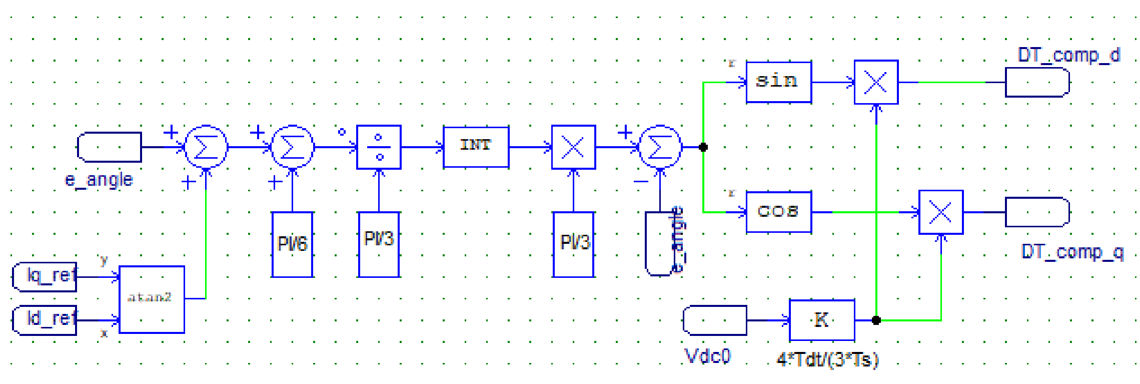

Figure 5.2.3.2: Dead Time Compensation

"INT" block truncates its input to the integer number to determine the sector of the current space vector in six sector space.

According to the equations, q axis dead time compensation voltage is added to the q axis voltage reference and d axis dead time compensation voltage is subtracted from the q axis voltage reference as shown in Fig.5.2.3.1.

5.2.4 Vector Tracking Observer Design

The schematic diagram of the vector tracking observer is shown in Fig.5.2.4.1. Controller gains are calculated according to the procedure explained in section 3.4.1.

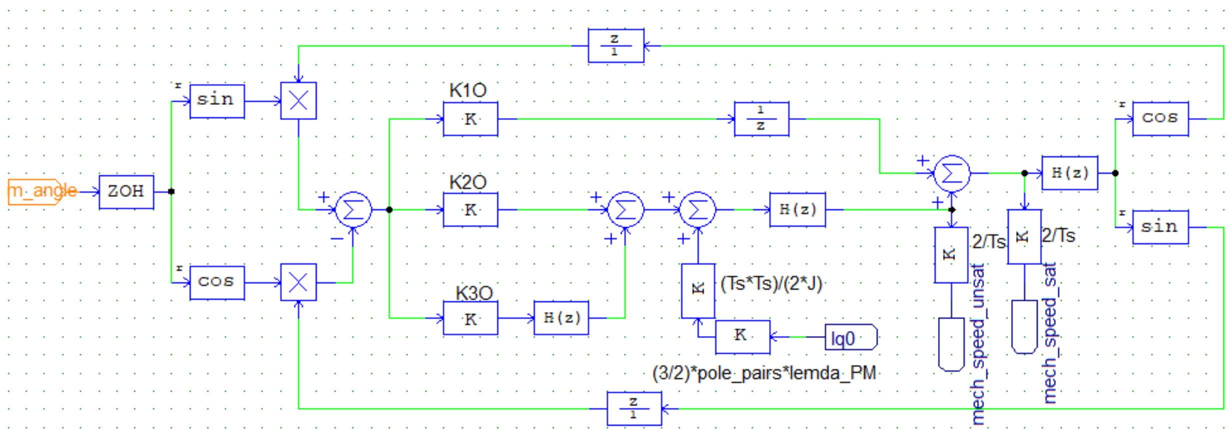

Figure 5.2.4.1: Schematic Diagram for Vector Tracking Speed Observer

Input to the observer is the mechanical angle of the rotor which is determined from the encoder output. Electromagnetic torque is added to the observer model as a feed-forward term.

Since the observer model is modified to simplify the calculations as discussed in section 3.4.1, the speed signals need to be multiplied by a factor of $2/T_s$ to obtain the actual speed values. Initially, the values of the feedback signals are unknown, thus it is necessary to delay the feedback signals by one sample period of time. Two unit delay blocks are used for this purpose and their initial output values are set to zero.

The VTO is designed to determine the mechanical speed of the motor, thus it can be directly used as the feedback for the speed control loop design. However, when calculating dq axis decoupling and back emf compensation terms it needs to be converted into electrical speed. As explained in section 3.4, both enhanced and un-enhanced speeds can be used. In our case, enhanced speed output is used because it gives an estimate of the actual speed without filtering.

5.3 Speed Control Loop

The schematic diagram of the speed control loop is shown in Fig.5.3.1. It is implemented using MTPA technique discussed in section 3.5.1. Controller outputs the reference torque command from which i_d and i_q references are determined using two look-up tables.

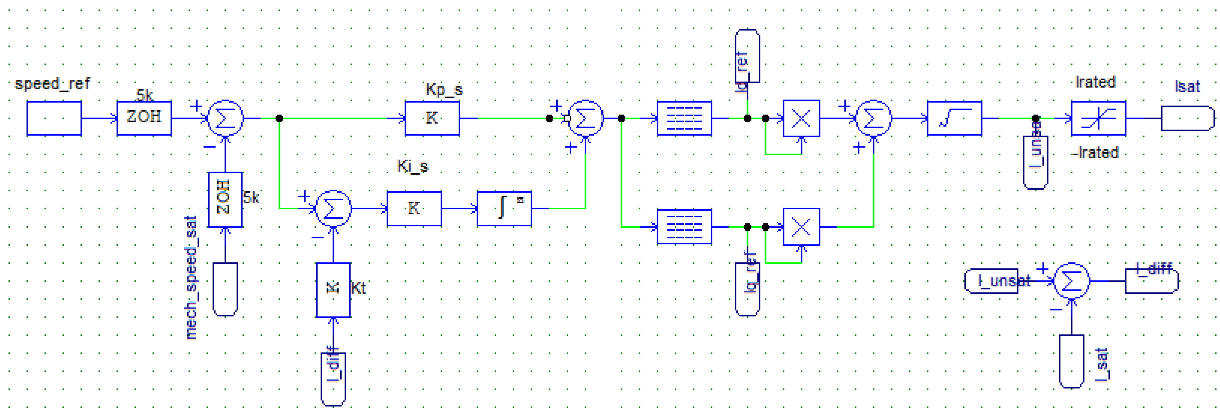

Figure 5.3.1: Schematic Diagram for Speed control Loop

Inputs to the speed control loop are sampled at 5kHz frequency which is slower than the current loop frequency of 10kHz. Two ZOH blocks are used to define the sampling rate for the speed signals. It should be noted that all the other ZOH blocks used in schematics have the sampling frequency of 10kHz.

Enhanced speed estimate of the VTO is used for the speed feedback to the controller. Anti-windup scheme is implemented considering the current reference saturation. Anti-windup gain for the speed controller is the torque constant (K_t) of the motor which is calculated from the ratio T_{rated}/I_{rated}.

5.4 Interrupt handling in SimCoder

When generating the code for a system that has multiple sampling rates, SimCoder uses the interrupts of the PWM generators for the PWM sampling rates. Thus, all the elements that are connected to the PWM generator and have the same sampling rate as the PWM generator, will be grouped together and implemented in an interrupt service routine in the generated code.

For other sampling rates in the control system, it will use the Timer 1 interrupt first, and then Timer 2 interrupt if needed, If there are more than three sampling rates in the control system, the corresponding interrupt routines will be handled in the main program by software. For blocks that do not have sampling rates associated with them, the corresponding code will be placed in the main program.

PSIM generated C code has the following functions (see Listing A.2 and A.3);

1. interrupt void Task (): The interrupt service routine for 10 kHz. It is called in every 10 kHz cycle.

2. interrupt void Task_1 (): The interrupt service routine 5 kHz. It is called in every 5 kHz cycle.

3. void Task_2 () : The function to start-stop PWM. It is called inside the main function.

4. void TaskS6 (): The function to implement rotor alignment at start-up, modelled inside subsystem S1 in the schematic

5. void TaskS8 (): The function to implement current control loop, modelled inside subsystem S2 in the schematic

6. void TaskS8_1 (): The function to implement speed control loop, modeled inside subsystem S2 in the schematic

7. void Initialize (): The initialization routine to initialize hardware.

8. void main (): The main program. It calls the initialization routine, and runs an infinite loop.

Separate functions are defined to implement subsystems S1 and S2 and they are called inside the respective ISRs. The header file "PS_bios.h" has been included to the c code. In this header file, PSIM functions are defined. All the functions start with "PS_" are PSIM defined functions. Ex : PS_EnableIntr ();

PWM switching frequency is 10kHz, so the 10kHz ISR uses the PWM clock timer and for 5kHz ISR, Timer1 is used. For initialization of Timer 1, see line 33-34 in Listing A.3.

Chapter 6

Experimental Results

6.1 Experimental Set-up

The experimental set up used to implement the control algorithm is shown in Fig.6.1.1. An SPMSM machine is connected to the IPMSM machine as a load. Output signals from the DAC and phase current measured using the current sensor are observed through an oscilloscope. Panic button is used to disconnect the voltage supply to the inverter when there is an unacceptable behavior in the control algorithm.

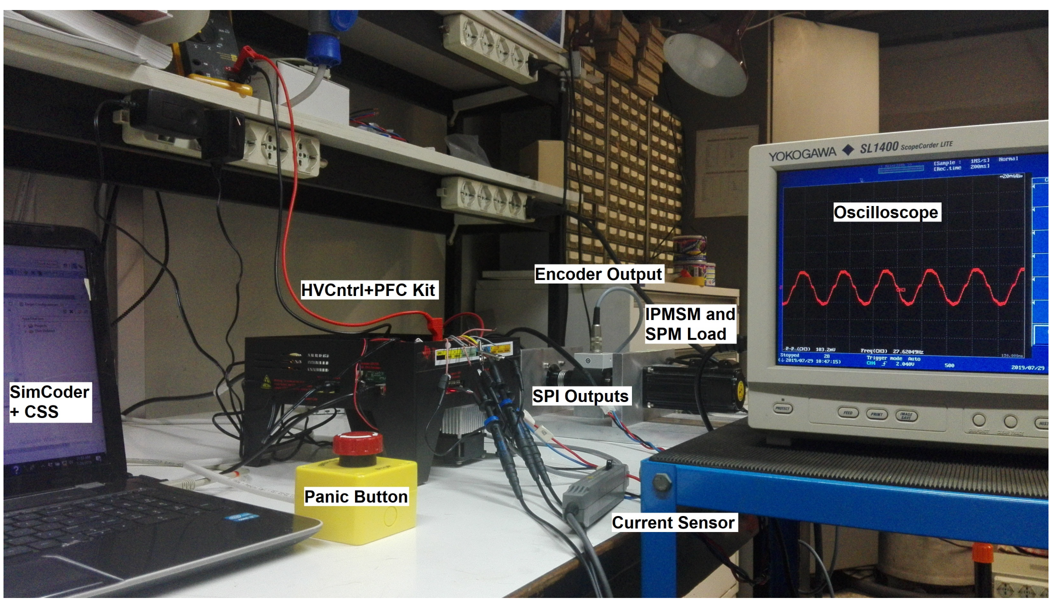

Figure 6.1.1: Experimental Set-up

6.2 Experimental results

6.2.1 Position and Speed Estimation from VTO

Fig.6.2.1.1 shows actual and estimated mechanical angle from the vector tracking observer. It is clear that VTO is able to track the angle nicely if it is tuned well.

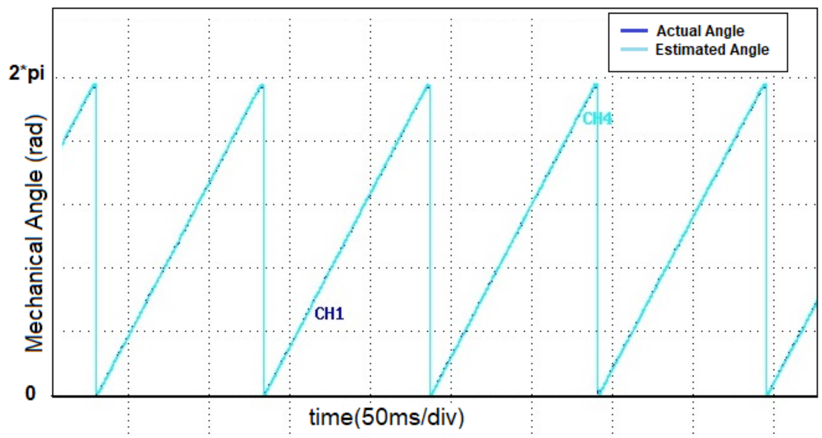

Figure 6.2.1.1: Vector Tracking Speed Observer Angle Estimation

Estimated mechanical angle and the speed (enhanced) are shown in Fig.6.2.1.2. Oscillations are observed in the estimated speed. It seems that the number of oscillations per revolution corresponds to the number of polar pairs of the machine. These oscillations get smaller when the motor is rotated at higher speeds.

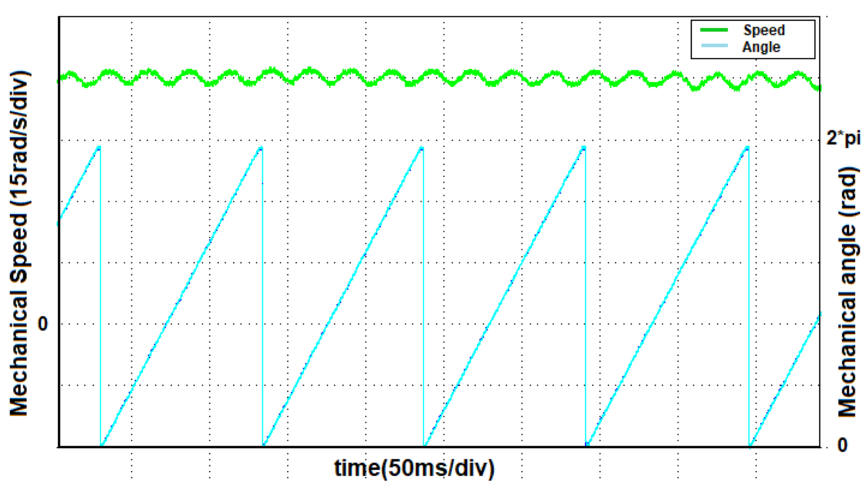

Figure 6.2.1.2: Vector Tracking Speed Observer Estimated Angle and Speed

6.2.2 Dead-time Compensation

As explained in section 3.3, due to the dead time introduced in the switches, the current signal gets distorted. The effect of dead time is shown in the phase current waveform in Fig.6.2.2.1.

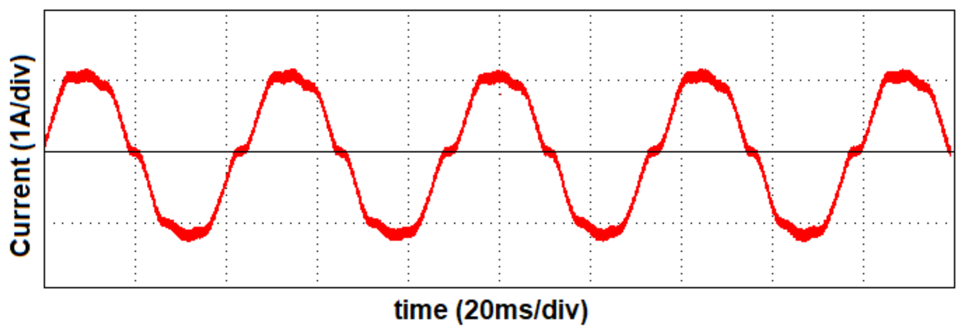

Figure 6.2.2.1: Effect of Dead-time on Phase Current Waveform

The dead-time compensation voltages are calculated in dq synchronous reference frame and added to the dq reference voltages for the inverter. The improved phase current waveform with dead-time compensation is shown in Fig.6.2.2.2.

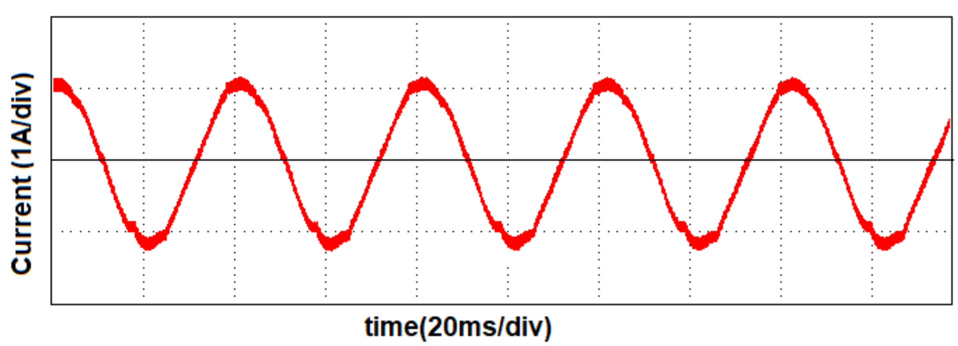

Figure 6.2.2.2: Phase Current Waveform with Dead-time Compensation

D and q axis dead time compensation voltages are shown in Fig.6.2.2.3.

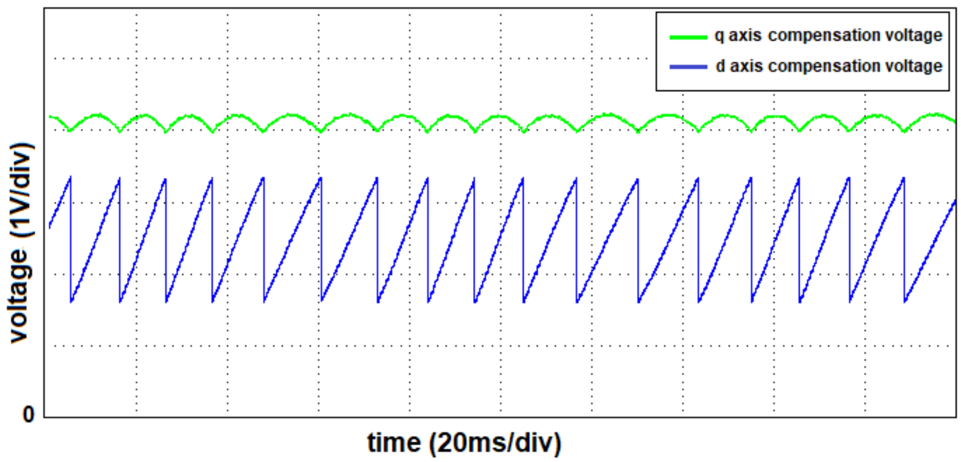

Figure 6.2.2.3: Dq Axis Compensation Voltages

When compensation voltages in dq reference frame are transformed into abc voltages, they are in phase with respective phase current waveforms. Fig.6.2.2.4 shows phase alignment in phase A compensation voltage and phase A current. Note that, in the figure, compensation voltage is scaled such that it has the same magnitude as phase current in order to clearly show their phase alignment.

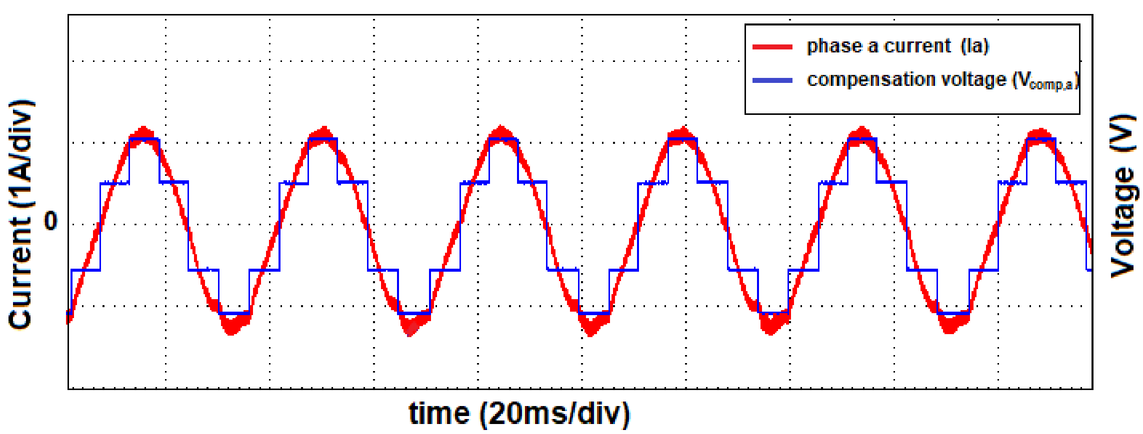

Figure 6.2.2.4: Phase Alignment in Compensation Voltage and Phase Current

Dead-time compensation voltages can also be added in alpha-beta stationary reference frame as voltages fed into the space vector modulation block are in alpha-beta reference frame. In that case, alpha-beta compensation voltages are shown in

Fig.6.2.2.5.

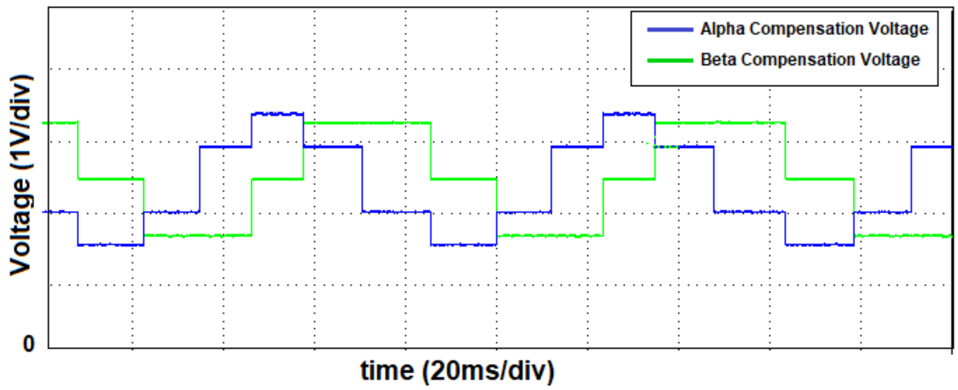

Figure 6.2.2.5: Alpha-beta Components of Compensation Voltages

6.2.3 Current Loop Testing

Cross-coupling decoupling current regulator is used for the experiments conducted to test control algorithms which are illustrated in the next subsections.

6.2.3.1 Current Loop Frequency Response

In order to verify the response of the current control loop, a sinusoidal i_d reference having frequency same as the current controller bandwidth (500Hz) is applied. Magnitude of the sinusoid is 0.2A and 0.4A DC offset is applied in order to avoid zero crossing. Q-axis current is set to zero so that the rotor is locked.

By convention at -3dB,

$$i_{d-meas} = \frac{i_{d-ref}}{\sqrt{2}} = \frac{200mA}{\sqrt{2}} = 141mA$$

$$i_{d-ref} - i_{d-meas} = 59mA$$

Input reference and the measured output currents are shown in Fig.6.2.3.1. The output is attenuated to a factor of 0.707 times the input (or -3dB) as expected.

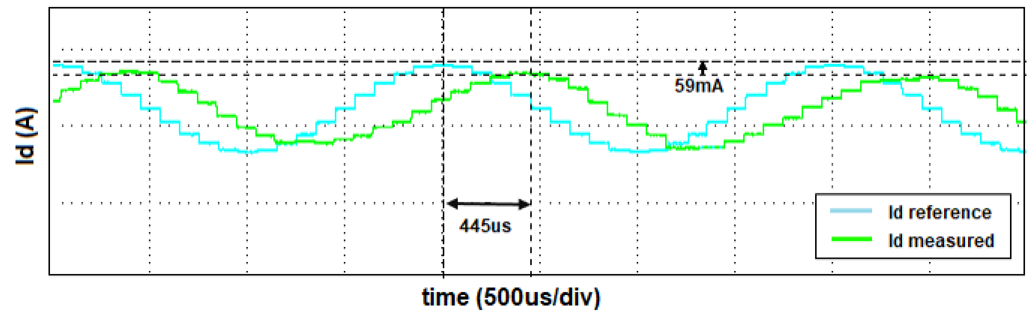

Figure 6.2.3.1: Input and Output d-axis Current Waveforms

The current loop bandwidth is kept at 500 Hz and the experimental data was obtained by changing the frequency of the input sinusoid. The difference between input and output is measured. In Fig.6.2.3.2, the experimental bode plot is plotted for 100% DC link voltage and 25% DC link voltage along with the theoretical bode plot which is obtained using the closed-loop transfer function given in Eq.3.1.2.6.

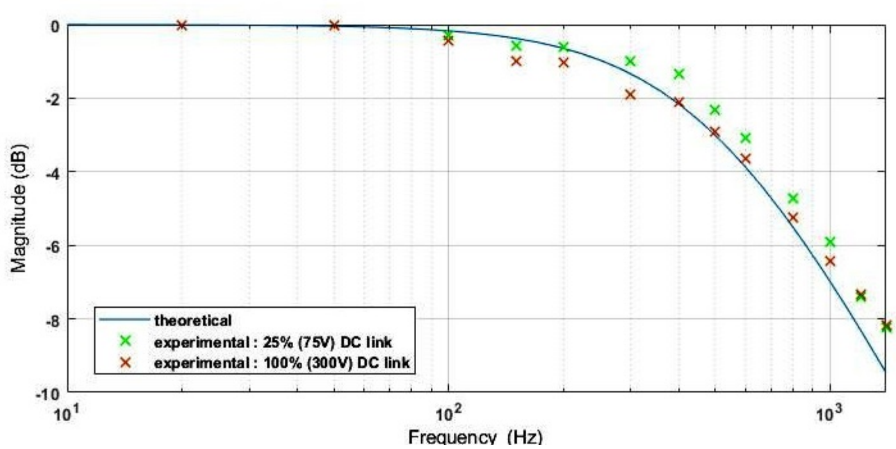

Figure 6.2.3.2: Frequency Response of Current Control Loop

The experimentally obtained frequency response is close to the theoretical one in both situations.

6.2.3.2 Current Loop Step Response

In order to verify the controller's ability to achieve set bandwidth, i_d response to a step-change in reference from 0.2A to 0.7A is observed for different bandwidths as shown in Fig.6.2.3.3 to Fig.6.2.3.5. Q-axis current reference is kept at zero. Time constant of the step response $\simeq \frac{1}{f_{BW}}$

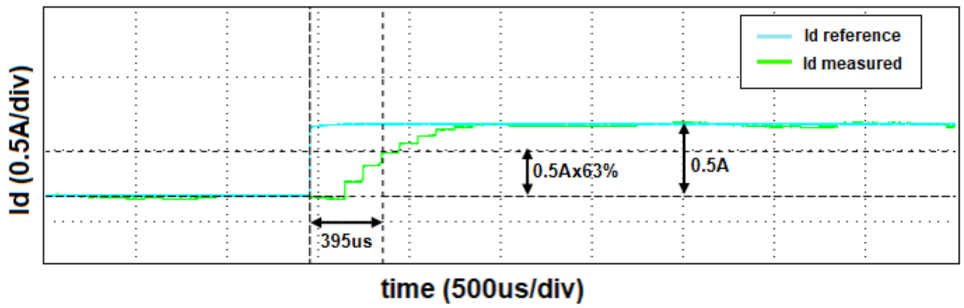

Figure 6.2.3.3: I_d Step Response at 500Hz Current Loop Bandwidth

$$\frac{1}{\tau} = \frac{1}{(395 \times 10^{-6})(2\pi)} = 403Hz$$

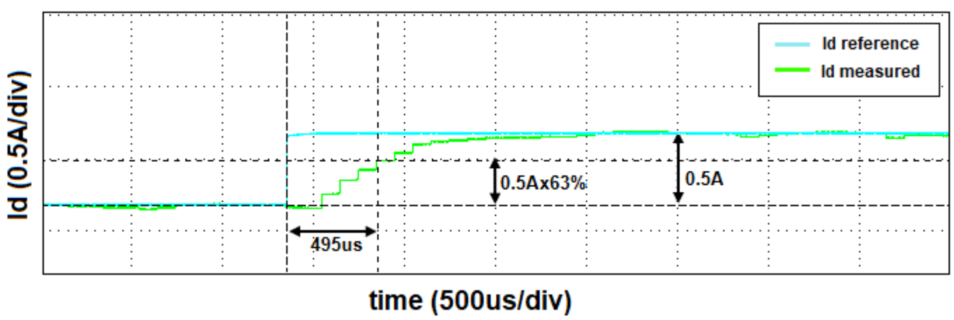

Figure 6.2.3.4: I_d Step Response at 400Hz Current Loop Bandwidth

$$\frac{1}{\tau} = \frac{1}{(495 \times 10^{-6})(2\pi)} = 322Hz$$

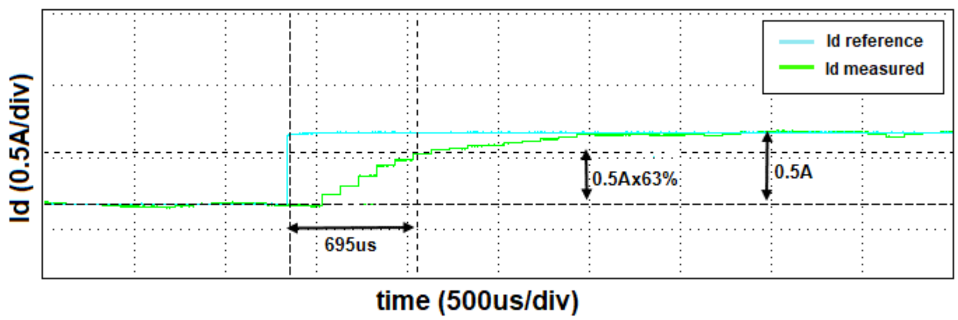

Figure 6.2.3.5: I_d Step Response at 300Hz Current Loop Bandwidth

$$\frac{1}{\tau} = \frac{1}{(695 \times 10^{-6})(2\pi)} = 229Hz$$

It is observed that the deviation is about 20% which acceptable since the relationship between the time constant and the bandwidth is an approximation. In this way, it is verified that the system response is close to the expected behavior.

Motor is then rotated by applying a q axis current reference. i_q reference is changed from 0.5A to 0.6A and i_d is kept zero. The step response of q axis current is shown in Fig.6.2.3.6 at 230 Hz current loop bandwidth.

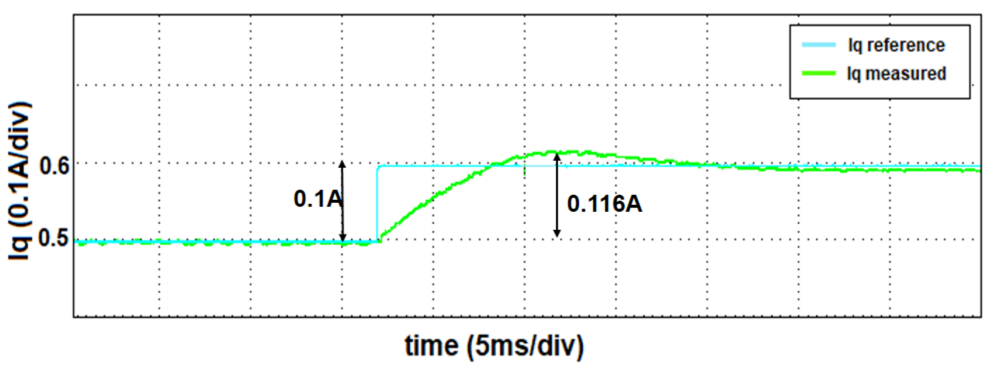

Figure 6.2.3.6: I_q Step Response at 230Hz Current Loop Bandwidth

Percentage of overshoot can be calculated as,

$$Overshoot = \frac{0.116 - 0.1}{0.1} \times 100\% = 16\%$$

Torque response to the above current step is shown in Fig.6.2.3.7.

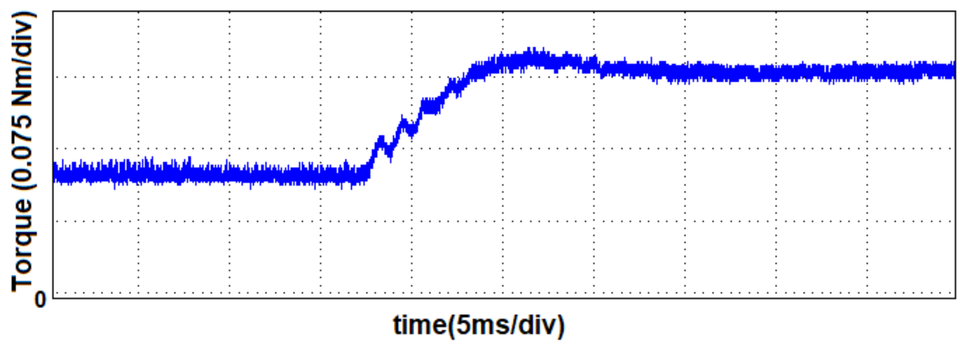

Figure 6.2.3.7: Torque Response at 230Hz Current Loop Bandwidth

6.2.4 Speed Loop Testing

Speed loop response is analyzed by changing speed and current loop bandwidths and the observer bandwidth. Fig.6.2.4.1 shows speed and phase current for 300Hz current bandwidth and 30Hz speed loop bandwidth. Fig.6.2.4.2 is for 500Hz current bandwidth and 50 Hz speed bandwidth. In both cases, observer bandwidth is 300Hz.

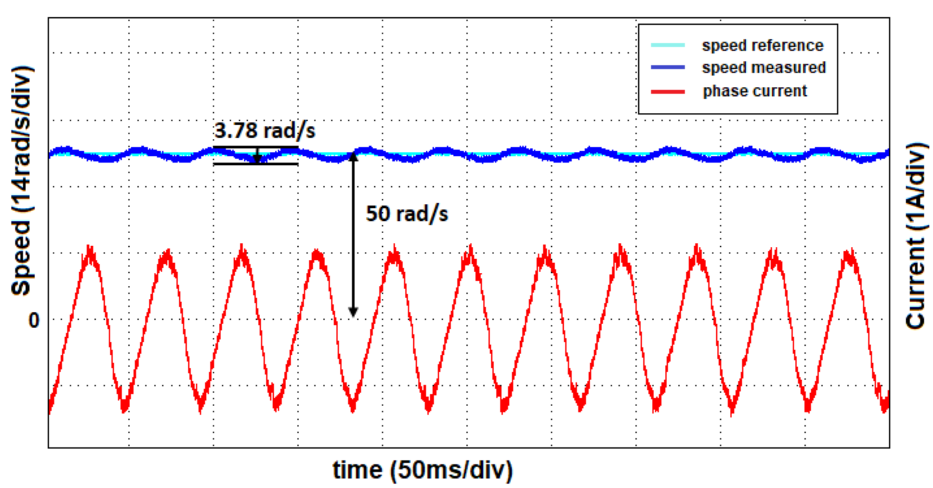

Figure 6.2.4.1: Speed and Phase Current at 300Hz Current BW and 30Hz Speed BW

Percentage of speed ripple can be calculated as;

$$Speed\ Ripple = \frac{3.78}{50} \times 100\% = 7.6\%$$

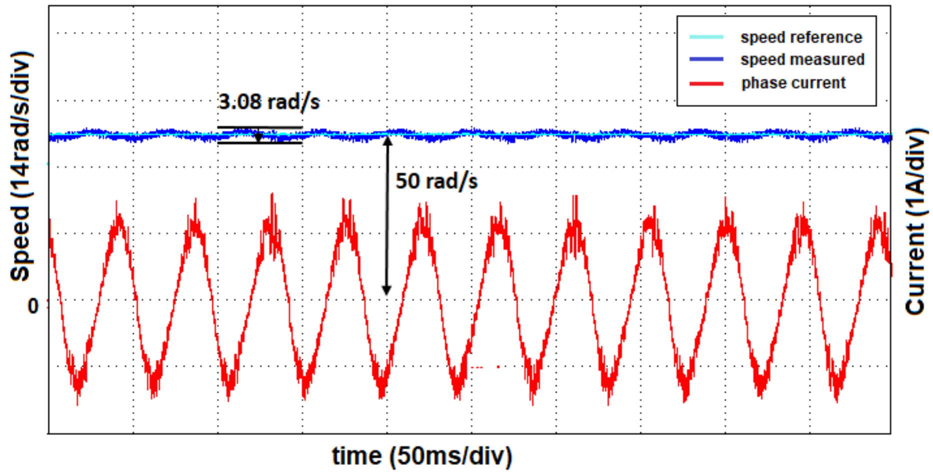

Figure 6.2.4.2: Speed and Phase Current at 500Hz Current BW and 50Hz Speed BW

Percentage of speed ripple;

$$Speed\ Ripple = \frac{3.08}{50} \times 100\% = 6.2\%$$

An oscillatory behavior is observed in the speed response and the period of the speed ripple is equal to the fundamental period of the current. The ripple in speed can occur due to the torque ripple produced by unequal strength permanent magnets. The ripple in the speed feedback obtained from the VTO can be another possible reason. From Fig.6.2.4.1 and Fig.6.2.4.2, it is observed that when speed and current loop bandwidths are increased, ripple in the speed gets reduced. However, it increases quantization noise in speed and current waveforms.

Fig.6.2.4.2 and Fig.6.2.4.3 show the step response observed for the two cases.

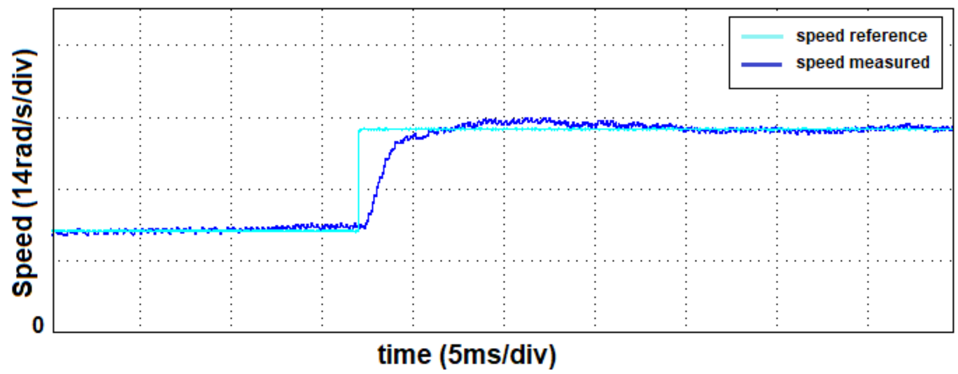

Figure 6.2.4.3: Speed Response at 300Hz Current BW and 30Hz Speed BW

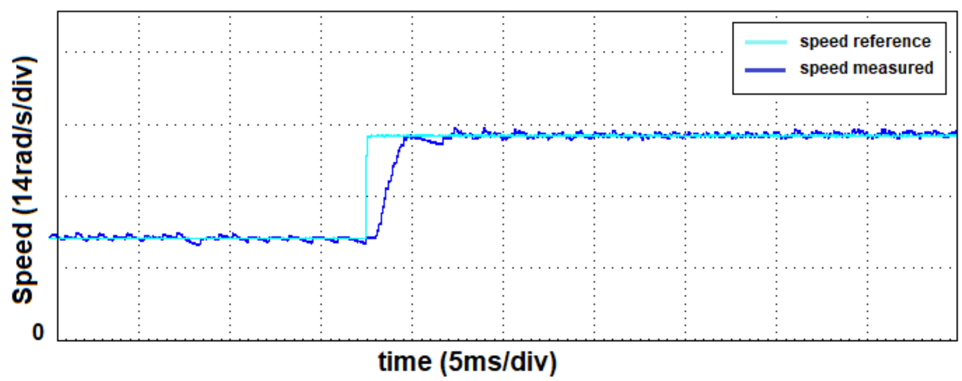

Figure 6.2.4.4: Speed Response at 500Hz Current BW and 50Hz Speed BW

It can be observed that changing speed and current bandwidths has not affected much to the response time. The oscillatory behavior can be seen in the step response which has 30Hz speed bandwidth. No overshoot is observed in the speed step response, since enhanced speed estimation of the VTO is used as the speed feedback to the controller.

Fig.6.2.4.5 and Fig.6.2.4.6 depict the speed and phase current waveforms when the observer bandwidth is 500Hz and 1000Hz respectively. In this case, the current loop bandwidth is 300Hz and the speed loop bandwidth is 30Hz. It is observed that increasing observer bandwidth results in high quantization noise in current and speed waveforms. A noticeable improvement in response time or speed ripple is not achieved when observer bandwidth is increased.

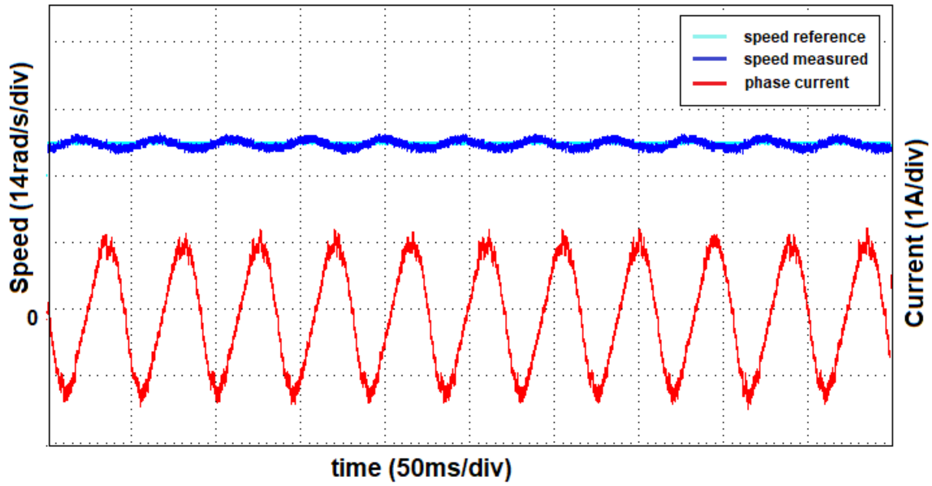

Figure 6.2.4.5: Speed and Phase Current at 500Hz Observer BW

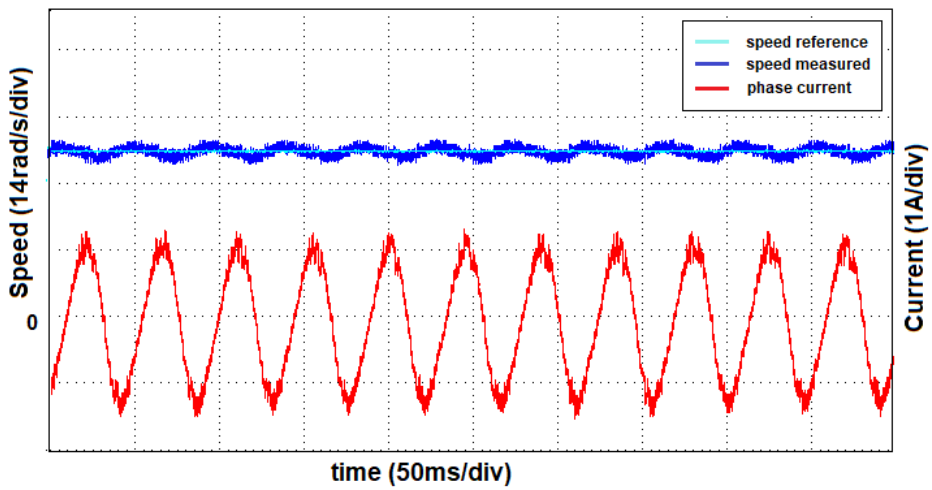

Figure 6.2.4.6: Speed and Phase Current at 1000Hz Observer BW

Chapter 7

Conclusion and Future Work

7.1 Summary and Conclusion

The key objective of this thesis is to develop and test a speed control algorithm for an IPMSM machine using PSIM software package based on vector control technique. The approach of using PSIM visual programming environment makes the implementation of motor drive control techniques easier and faster than conventional C programming.

The inner current control loop is designed with dq axis decoupling and dead time compensation. Two types of current regulators are considered; Cross-Coupling Decoupling Current Regulator and Complex Vector Current Regulator. An anti-windup scheme for the speed controller is implemented considering the voltage saturation. For closed-loop speed control, back-emf compensation and dq axis decoupling, estimation of speed of rotation of the motor is required. The Vector Tracking Observer (VTO) is designed to estimate speed from the rotor position given by the incremental encoder.

Since the position provided by the incremental encoder is not absolute, initial rotor alignment procedure is implemented to bring the rotor to its zero position at start-up.

Frequency response of the current control loop is obtained by applying a sinusoidal d-axis current reference at different frequencies. Furthermore, The ability of the controller to achieve set bandwidth is verified by applying step current references at different current loop bandwidths.

The speed control loop is designed based on the Maximum Torque per Ampere (MTPA) technique, in which reluctance torque is utilized to have a maximum torque for a given current. For the speed controller, current reference saturation is considered when designing anti-windup scheme.

Finally, the performance of the speed loop is tested with changing speed and current loop bandwidths, and observer bandwidth. A ripple in speed is observed and the ripple becomes smaller for higher speed and current bandwidths. However, at higher speed and current bandwidths, more noise in the responses are observed.

7.2 Future Work

This thesis work can be extended to design position controllers for motor drives. The parameters that affect the performance of the controller design can be investigated further, to improve the system response. Moreover, the influence of position sensor resolution on the performance of the drive can be experimentally investigated.

The load machine drive control can also be implemented in order to have a fully functional and regenerative test bench in which one machine acts as a speed-controlled motor and the other as a torque-controlled load.

High-speed operation control techniques such as flux weakening control can be implemented based on this thesis work. Moreover, advanced PWM techniques at operational limits can be implemented and tested.

Appendix A

C Code

```
static float F_sw;
static float T_sw;
static float time;
static float V;
V=x1;
F_sw=10000;
T_sw=1/F_sw;

time = time+T_sw;
if (time>1.5)
{time=0;}

if ((time>0)&&(time<(0.5)))
{ y1=V;
y2=0;}

if ((time>0.5)&&(time<(1.0)))
{ y1=-V*0.5;
y2=V*0.866;}

if ((time>1.0)&&(time<(1.5)))
{ y1=-V*0.5;
```

```
y2=-V*0.866;}
```

Listing A.1: C code Block for Generating Alpha-beta Voltages for Rotor Alignment

```
interrupt void Task();        //ISR for current control loop
interrupt void Task_1();      //ISR for speed control loop
void Task_2();                //Start-Stop PWM
void TaskS6(DefaultType *fOut0, DefaultType *fOut1, DefaultType *
    fOut2); //rotor alignment at start-up
void TaskS8(DefaultType fIn0, DefaultType fIn1, DefaultType fIn2,
    DefaultType fIn3, DefaultType fIn4, DefaultType *fOut0,
    DefaultType *fOut1, DefaultType *fOut2);
void TaskS8_1();
```

Listing A.2: Functions in PSIM generated C Code

```

void Initialize(void)
{
  PS_SysInit(30, 10);
  PS_StartStopPwmClock(0);  // Stop Pwm Clock
  PS_InitTimer(0, 0);

  // Set initial states for those GPIO/AIO output ports.
  PS_ClearDigitOutBitA((Uint32)1 << 22);  // Reset GPIO22
  PS_InitDigitOut(22);  // for measuring the execution time of 10000
    Hz interrupt routine

  PS_ResetAdcConvSeq();
  PS_SetAdcConvSeq(eAdcCascade, 7, 0, 1.0);
  PS_SetAdcConvSeq(eAdcCascade, 9, 1, 1.0);
  PS_SetAdcConvSeq(eAdcCascade, 11, 2, 1.0);
  PS_AdcInit(1, !2);

  PS_InitPwm3ph(1, 3, (double)10000*1, (2E-6)*1.e6);  // pwnNo,
    waveType, frequency, deadtime
  PS_SetPwm3phPeakOffset(1, 2, (-(1.0)), 1.0/2);
  PS_SetPwm3ph1AdcIntr(ePwmIntrAdc, 1, 0);
```

```
  PS_SetPwm3ph1Vector(ePwmIntrAdc, Task);
  PS_SetPwm3ph1TzAct(eTZHighImpedance);
  PS_SetPwm3ph1UvwSL(0, 0, 0);
  PS_StopPwm3ph1();

  PS_InitEncoder(1, 1, 0, 0, 0, 0, 10000);

  PS_InitEncIndexStrobeLatch(1, 0);

  PS_SpiInitBBR(PSA_SpiDev, Csz_SPI_DEVICES);
  PS_SpiInit(PSA_SpiInOut, PSA_SpiQueue, Csz_SPI_QUEUE, 1);

  PS_InitTimer(1,30000L);
  PS_SetTimerIntrVector(1, Task_1);
  PS_StartStopPwmClock(2);  // Start Pwm Clock, start Timer1
}

void main()
{
  Initialize();
  PS_EnableIntr();   // Enable Global interrupt INTM
  PS_EnableDbgm();
  for (;;) {
    PS_SpiTransmitCheck();
    Task_2();
  }
}
```

Listing A.3: Initialization and Main Functions

Bibliography

[1] Harshada V. Deo, International Journal of Innovative Research in Technology, Vol.11(1), 2014 *A Review of Speed Control Techniques Using PMSM*

[2] *Field Oriented Control of Permanent Magnet Synchronous Motors User's Guide*, Microsemi

[3] Sang-Hoon Kim, *Electric Motor Control DC, AC and BLDC Motors*, chapter 4.2. Elsevier, 2016

[4] Adam Walker, Lecture slides:*Permanent Magnet Motor Drives*, University of Nottingham

[5] N.Kiran, A.Nair, D.Sri Lakshmi, International Journal of Advanced Research in Education and Technology, Vol.3(4), Oct-Dec 2016, *Permanent Magnet Synchronous Motor Control with Speed Feedback Using a Resolver*

[6] Ralph Kennel, Lecture notes:*Space Vector Modulation*, Technical University of Munich

[7] Kaushik Jash, Int. Journal of Engineering Research and Applications, Vol.3(5), 2013, *Vector Control of Permanent Magnet Synchronous Motor Based On Sinusoidal Pulse Width Modulated Inverter with Proportional Integral Controller*

[8] Paul Krause,Oleg Wasynczuk, *Analysis of Electric Machinary and Drive Systems*, chapter 3. IEEE, 2013

[9] Seung-Gi Jeong, IEEE Transactions on lndustrial Electronics, Vol.38(2), April 1991, *The Analysis and Compensation of Dead-Time Effects in PWM Inverters*

[10] Takashi Sukegawa, IEEE Transactions on Industry Applications, Vol.27(3) May-June 1991,*Fully Digital, Vector-Controlled PWM VSI-Fed ac Drives with an Inverter Dead-Time Compensation Strategy*

[11] Giulio De Donato, PhD thesis, 2006 *Implementation and Performance Evaluation of a Low Cost Surface-Mounted PM Machine Drive Using Binary Hall-effect Sensors*, Sapienza University of Rome

[12] *High Voltage Motor Control and PFC Kit Hardware Reference Guide*, C2000 Systems and Applications Team, Texas Instruments.

[13] *SimCoder User's Guide*, Powersim

[14] Powersim Tutorial,*Using SPI in F2833X/F2803X Target*

[15] Fathima Razzaq, Masters thesis, 2018 *Complex Vector Synchronous Frame PI Current Regulator Development in Discrete-time Domain*, University of Oviedo

[16] Datasheet, *AD7568 Digital to Analog Converter*

[17] Sinan, Mehmet Timur, *A novel initial rotor position alignment method for permanent magnet synchronous motor using incremental encoder*